

# An Artificial Intelligent based System to Automate Decision Making in Assembly Solution Design

ABADI Chaimae<sup>1</sup>, MANSSOURI Imad<sup>2</sup>, ABADI Asmae<sup>3</sup>

Laboratory of Mechanics, Mechatronics and Command, Electric Power, Maintenance and Innovation Team

Ecole Nationale Supérieur d'Arts et Métiers (ENSAM-Meknès)

Moulay Ismail University, B.P. 4042, 50000, Meknes, Morocco<sup>1,2</sup>

INSA, Euro-Mediterranean University of Fez, Fes, Morocco<sup>3</sup>

**Abstract**—Nowadays, competitiveness between industries has become very strong. Thus, industries are faced to serious challenges in terms of products qualities, time development and production cost. As assembly operations difficulties cause a big part of production problems, the integration of assembly selection since the earlier product life cycle phases has become a necessity for every company in order to survive. However, despite the large number of approaches that have been proposed in order to achieve this integration goal, many other problems are still present. It is in this context that a flexible and automated decision making system is proposed. It is based on ontologies and also on the Case Based Reasoning (CBR) and Rule Based Reasoning (RBR) concepts. Indeed, this system is an automation of the integrated DFMMA approach, in particular its assembly solution selection methodology. The developed system permits to designers avoiding the redundancy in the works by benefiting from their previous studies and their experience. In addition to that, it facilitates and automates the assembly solution selection even if the number of assembly alternatives is high. Finally, to illustrate the efficacy of the proposed system, a case of study is developed in the end of the work.

**Keywords**—Assembly selection; product assembly; ontologies; CBR & RBR; flexible and automated; decision making system; artificial intelligence

## I. INTRODUCTION

Nowadays, competitiveness between companies and their industrial products has become very strong. Actually, customer's requirements, development speed of the world and its continuous changes face industries to high levels of challenges in order to survive in the market and to keep in their places. Thus, industrial companies have to be quick and efficient in the same time in the development and the production of their products.

As assembly operations occupy a very important place in this context and their success without difficulties help in a high percentage to the achievement of any industrial company goals, this production phase has to be taken into consideration since the earlier stages of product life cycle, namely in the design phase. It is in this context that the DFA (Design For Assembly) concept and many other adjacent concepts have appeared. However, the majority of their approaches present two main limits. Firstly, they are theoretic and manual, thus in case of having a high number of assembly alternatives, the selection becomes impossible. Secondly, the experience of designers is not taken into consideration which produces a

redundancy in the work. Consequently, a lack of selection quality and also a waste of time and money are present.

To overcome all those limits, an automation of the optimal assembly solution selection becomes an obligation.

Thus, in this paper, a flexible and automated assembly decision making system based on ontologies is developed.

Indeed, through the high capacities and the different potentialities of this emergent artificial intelligent tool, the selection of the optimal assembly solution since the product design phase could be done in an efficient and automated way. In addition to that, in order to achieve the same goal, the proposed automated system is based on two main types of reasoning which are: the CBR (Case Based Reasoning) and the RBR (Rules Based Reasoning).

The approach, that was chosen to be automated, is the assembly solution selection methodology of the DFMMA (Design For Materials, Manufacturing and Assembly) approach. This choice is done because of the different advantages that present this integrated design approach. All its details are presented in our previous work, Reference [1] but the implementation of the approach was not automated. Thus, this work is an automation of the assembly selection part of our DFMMA approach.

It is to note that the proposed ontology is named ADM-Onto referring to Assembly Decision Making Ontology. With regards to the proposed system, it calls ADM system which refers to Assembly Decision Making system.

Thus, in the next section, a literature review about the different DFA approaches, about the assembly methodology presented by the DFMMA approach and its advantages and also about ontologies is presented. In the third section of this paper, the ADM system is described. The ADM-Onto is constructed in the fourth section. Section five and six are dedicated to the description of the working process of the CBR and the RBR modules of the automated ADM system. Finally, to illustrate the efficient results of the ADM Methodology (Assembly Decision Making Methodology) a case of study is developed at the end of the work.

## II. LITERATURE REVIEW

The first phase of any product life cycle is the design. It plays a key role in the optimization of the product quality, its production time and its cost later.

According to the systematic approach proposed by Pahl and Beitz [2], this phase is composed of four main stages which are: Product planning and clarifying the task, Conceptual design, Embodiment design and Detail design.

Each one of those stages aims to have a set of specific objectives that contributes to the achievement of the next stage.

Having regard to the detail design phase, one of its main goals is to select the optimal assembly solution for the studied product. It is in this context that Design For Assembly (DFA) has appeared and many approaches have been proposed to support the assembly phase from the earlier phases of product life cycle.

One of the most famous proposed approaches is the Hitachi method, called also AREM (Assembly Reliability Evaluation Method) [3]. It aims to determine the different kinds of faults that can take place during the assembly phase of a complex product. It is based on two indicators: an assemblability evaluation score ratio (E) that estimates design quality from the difficulty of operations and an assembly cost ratio (K).

Another DFA approach is the Lucas method [4] that bases its optimal assembly solution selection on three indexes related to three separated and sequential analyses: the functional analysis, the feeding analysis and the fitting analysis.

Moreover, Boothroyd and Dewhurst have also proposed a practical method [5] that has as an objective the comparison between different assembly alternatives basing on their manual assembly rate. This rate is defined from an efficiency indicator ( $E_m$ ) that can be calculated using the following formula:

$$E_m = \frac{T_m}{T_a} = \frac{N_m \times 3}{T_a} \quad (1)$$

With:

- $T_m$ : the ideal time for assembly,
- $T_a$ : the real time to make the operation
- $N_m$ : the ideal number of product parts

In this index ( $E_m$ ) formula, the ideal time is correlated to the ideal number of parts by considering 3 seconds for their assembly.

In the same context of DFA, Samy and ElMaraghy have developed a methodology based on the product assembly complexity as an index to optimize the assembly solution selections [6,7].

In contrast to all those traditional works that base their assembly solution choice on different indexes, many new researches have treated the DFA using other different ways. Actually, Stone and McAdams have proposed a DFA approach based on the functional basis concept and the method of module heuristics [8]. Furthermore, Favi and Germani have developed a flows analysis based DFA method

[9]. In another work, Favi et al. have taken into account several aspects such as assemblability, manufacturability and costs in order to perform the selection of the best product modules configuration [10].

In addition to the DFA concept, many other concepts have appeared to support assembly problems from the earlier stages of product life cycle (in the design phase) namely, the DFMA (Design For Manufacturing and Assembly) concept and the integrated approaches of design [11].

One of the new relevant approaches, that have been developed, is the integrated DFMA approach [1]. It is based on the systematic approach of design and integrates in the same time the most important pillars of product life cycle, which are: design requirements, materials characteristics, manufacturing parameters and the assembly process specifications. It is based also on different quantitative indexes and tools without neglecting the subjective side of design optimization problem.

In regards to the selection of the optimal assembly solution methodology that proposes the DFMA approach, a quantitative analysis strategy is developed and the decision is based on three indicators which are related to the three basic notions of any product lifecycle: the product quality, the product lifecycle time and the product cost.

Actually, by calculating a quality indicator, a time indicator and a cost indicator, a global assembly index can be calculated. The optimal solution is the one that has the highest value of this global indicator.

Table I presents the different proposed formula of those indicators [1].

Despite the advantages of each of those works, they present two main common limits. Firstly, any one of those proposed approaches takes into account the experience of the design team. Actually, designers do not benefit from their previous design studies, in particular, the optimal assembly selection of the ancient studied products. In addition to that, all those approaches are theoretical and manual. Thus, if the design problem is composed of a high number of assembly alternatives, the comparison between them becomes difficult and sometimes impossible. Then, those two limits present a serious problem in terms of quality decision-making and a big waste of time and money.

To overcome those problems, a flexible and automated assembly decision making system is proposed in this paper. It aims to automate the selection of the optimal assembly solution and to permit to designers the use of their ancient studies with an automatic way. To do so, an emergent artificial intelligent tool is used, namely, inference ontologies.

Indeed, an ontology is an explicit and formal specification of the concepts, individuals and relationships which exist in some area of interest. It is built by defining axioms that describe the properties of these entities [12, 13].

TABLE I. SUMMARY OF ASSEMBLY SOLUTION SELECTION INDICATORS PROPOSED IN THE DFMMA APPROACH [1]

Indicator	Formula	Notation
The Quality Indicator	$I_Q = \frac{N_{sp.ver}}{N_{sp.tot}}$	N <sub>sp.ver</sub> : number of specifications which are verified by the assembly alternative. N <sub>sp.tot</sub> : total number of the imposed specifications
The Time Indicator	$I_T = \frac{f_{fa} \times f_{am} \times f_{ir} \times T_{id}}{T_{re}}$	f <sub>fa</sub> : Fatigue factor f <sub>am</sub> : Ambient factor f <sub>ir</sub> : Irregularity coefficient T <sub>id</sub> : ideal/ theoretical time to assembly the studied product T <sub>re</sub> : real time to assembly entirely the studied product
The Cost Indicator	$IC = 1 - \frac{C_{AT}}{C_{av.pro}}$	C <sub>AT</sub> : total assembly cost of the alternative C <sub>av.pro</sub> : total average cost of production of the studied product
The Global Assembly Indicator	$IG = IQ \times IT \times IC$	-----

Ontologies have different capacities [14]. The first one is their integration and completeness, assured by language expressivity [15]. In addition to that, they are characterized by an embedded intelligence, due to their reasoning capabilities [16, 17]. Finally, they offer a dynamism and flexibility abilities through queries and web services.

Thanks to all those cited potentialities, Ontologies are used in different domains, namely: the supply chain management [18 - 20], the Product Lifecycle Management [20, 21], the collaborative Product Development and Simultaneous engineering [22]. Actually, in the context of industrial product design phase, many works have been developed as examples:

- The proposed Product Design ontology (PDO) [20]: It supports collaboration between designers
- The Ontology Decision Support ontology (DSO) [23]: It supports decision making in the collaborative design of the product
- The ontology-based model [24]: It provides understanding and semantic interoperability between the different partners in the context of collaborative products development and it ensures the capitalization of the previous projects knowledge
- The developed ontology of Bock et al. [25]: It consists on representing the different possible designs for a same product in order to have an agreement between the various partners on the architecture of the product from the early phases of its life cycle.
- The ontology proposed by Mostefai et al. [26]: It represents three basic points of view in Collaborative Product Development which are the design of the components, the assembly and the production plan.
- The developed ontology of Lee et al. [27]: It is a meta-ontology for products' design. It is based on five root concepts: Attribute, behavior, entity, property and object relationship.
- The ontology proposed by Kim et al. [28]: It supports the collaborative design of products
- The proposed ontology of Chang et al. [29]: It supports data integration and decision making during the collaborative design of products.

All those works have used ontologies either to represent product information or to represent its development process data. Some of them have combined the two in a clear and generic way [30]. In addition to that, those previous researches have not exploited the reasoning capacity of ontologies which offers the ability to infer new information.

Thus, in this paper, the proposed decision making system focus on the overcoming of this limit. Indeed, unlike many of the existing works, it uses this fundamental capacity, namely, the reasoning ability, that ontologies offer in order to make the optimal assembly solution selection of complex mechanical products easy and automated.

### III. THE GLOBAL PROPOSED ADM METHODOLOGY

The goal of this paper is to automate the assembly selection phase. To do so, a structured methodology is proposed. It is based on ontologies, Case Based Reasoning (CBR) and Rule Based Reasoning (RBR) tools and methods. In fact, the combination of those three tools gives birth to a flexible and automated assembly decision making (ADM) system. In addition to that, the proposed methodology is based on the integrated DFMMA approach [1]. The choice of this approach was not random, but because of its different advantages, notably, the use of different quantitative indexes and tools throughout the study without eliminating the subjective side of design optimization problem.

Fig. 1 shows the global structure of the ADM system. According to the DFMMA approach, after the embodiment design that gives the design team its global architecture as a result, the studied mechanical product is divided into a set of modules which are assembled between them across different interfaces [1]. This step is done in the context of detail design phase. The second step of this phase is the materials and manufacturing processes selections. After this, the assembly solution selection should be done using the results of the previous steps as inputs.

As it is shown in Fig. 1, the proposed structured ADM system is decomposed to seven modules:

- The Information Collection Module (MCM): It is the first module in the proposed system. In fact, assembly selection step cannot be done correctly without sufficient information about the studied mechanical product. All of that information are the outputs of the previous steps of the design process.

- The Assembly ontology and the semantic model creator: Through the different characteristics and capacities of ontologies in expressiveness and storage, all the information collected in the previous module are expressed in a uniform language and capitalized in a same data base, which facilitates their use in the next stages. In addition to that, through this artificial intelligence tool, they are translated to a semantic model which will be the base of the subsequent decision-making.
- The Pre-Judging Module (PJM): This module permits to the design team the classification of the mechanical product in terms of components number. Actually, if the studied mechanical product is composed of one single element, it doesn't need to be assembled. In this case, the process of the proposed methodology is stopped. Otherwise, the process continues and the CBR/RBR processes are applied.
- The Case Based Reasoning Module (CBR module): This module allows the use of the previous assembly selection studies stored in the assembly ontology (the ADM-Onto) of the design team. In fact, the constructed assembly ontology contains a case base of the ancient mechanical products design. Thus, the design team uses their ancient studies and benefits from its experience to select the optimal assembly solution of its studied case. Actually, the studied mechanical product is adapted to a similar one, either by adopting it completely or by modifying its dissimilar parts.
- Rule Based Reasoning Module (RBR module): This module is responsible on the execution of reasoning rules elaborated by the design team in the constructed assembly ontology. These rules are based on the decision making process of the integrated DFMMA approach, in particular its assembly solution selection process. Moreover, they are expressed in SWRL (Semantic Web Rule Language)/SQWRL (Semantic Query-Enhanced Web Rule Language) and they are stored in the rules base of the developed assembly ontology.
- The Final Decision-Making Module (FDMM): This module permits to the design team to save and archive the generated results, namely, the optimal assembly solution in the case base of the ADM-Onto. So that, the designers benefit of their experience in the next studies in terms of time and consequently of money.
- The Control Module (MD): This module is responsible on the control of the consistency between all the other modules.

Fig. 1 shows the global structure of the proposed automated system for assembly solution selection.

Fig. 2 presents the working process of the proposed assembly decision-making system: the ADM system.

Indeed, after the collection of the different information that designers need to continue their study through the first module of the proposed system, the semantic model of the studied

mechanical product is created. It presents the base for the PJM to decide if the process of the proposed methodology continues or not. If the result is positive, then, the CBR based decision making process is executed through the CBR module. In its turn, if the optimal assembly solution is not got, the RBR based decision making process is executed. Finally, the generated result is saved through the FDMM.

In the next sections, the working processes of the proposed system modules will be detailed.

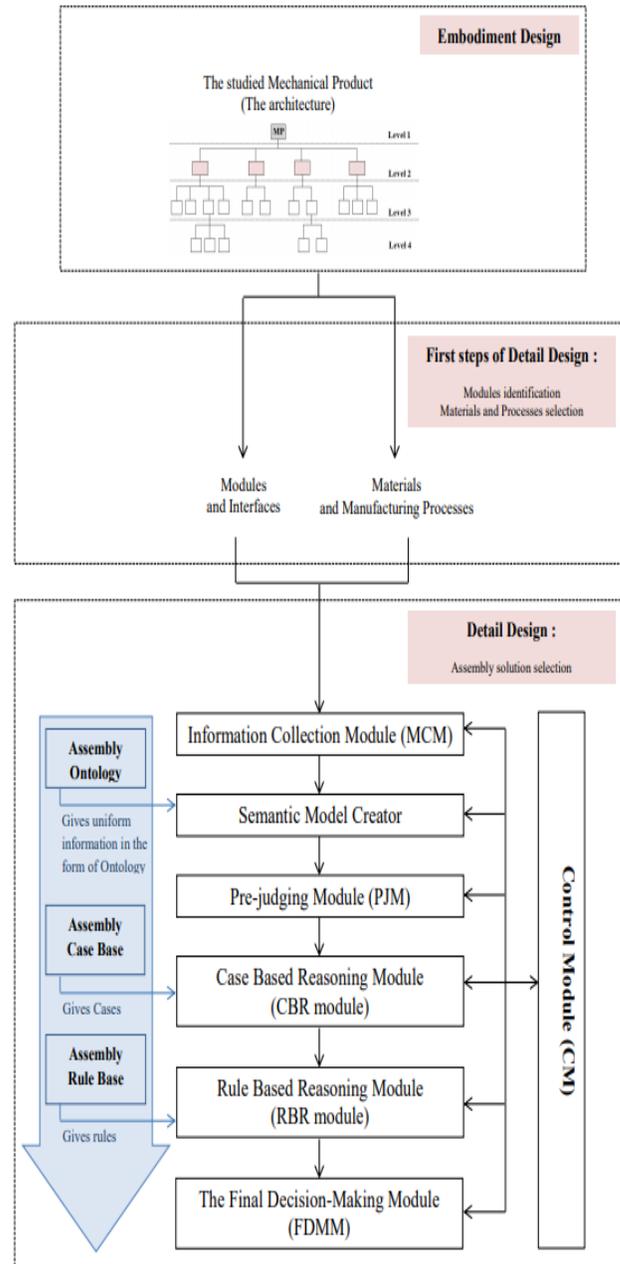


Fig. 1. The Global Structure of the Proposed Automated System for Assembly Solution Selection.

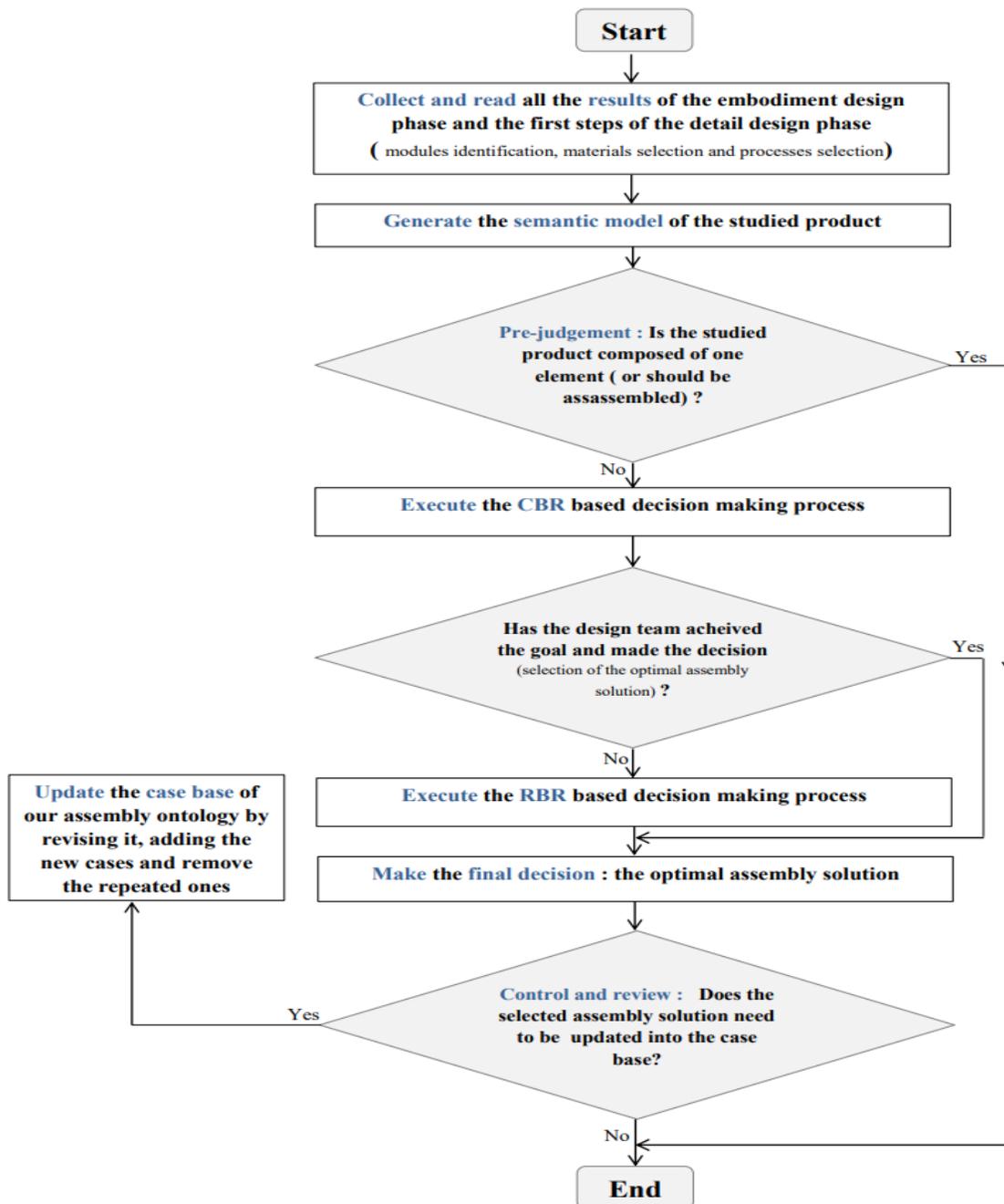


Fig. 2. The Working Process of the ADM System.

#### IV. CONSTRUCTION OF THE ADM-ONTO AND THE PRODUCT SEMANTIC MODEL

In this section, the assembly ontology (the ADM-Onto) is constructed to be used later by the CBR and RBR modules. In addition to that, an appropriate semantic model to the proposed ontology is developed.

##### A. Construction of the ADM-Onto

In this paper, the ADM-Onto is constructed in the Protégé ontology editor. Consequently, it is composed of three principal elements which are [31, 32]:

- **Classes:** They are defined as a set of individuals in a specific domain. The considered domain in our case is products assembly.
- **Object properties:** They define relations between classes and individuals.
- **Data properties:** They define modifiers for ontology classes or establish characteristics of the instances.

Thus, the first step to construct the ADM-Onto is to define its different classes as it is shown in Fig. 3. Each class except root classes (as an example: "*MechanicalProduct*" class) can

be a subclass of another one. One of the originalities of the proposed ontology classes consists on the definition and use of the "AssemblyIndicator" class which smooth the characterization of each assembly solution (assembly alternative) with quantitative indicators. Consequently, the comparison between alternatives becomes quantified.

The definition of Object properties is the second step of the assembly ontology construction. Table II presents those different object properties, their domains, their ranges and their inverse properties.

The defined object properties can be divided into three categories:

- Object properties 1-9: they are used to define the architecture of the studied mechanical product and the different requirements that should be respected in the design study. They are used mainly in the construction of the first part of the semantic model.
- Object properties 10-17: They are used to specify the different characteristics of the mechanical product components in terms of topologies, materials and manufacturing processes. Thus, they are used in the

CBR process in order to facilitate the research of the similar existing cases to the studied one.

- Object 18- 33: They are used to affect to each assembly solution the different indexes defined in the integrated DFMA approach, in particular the assembly solution selection part; namely the quality indicator, the time indicator, the cost indicator and finally the global assembly indicator [1]. Consequently, they are used in the RBR Process in order to facilitate the comparison between different assembly alternatives.

The third step of the ADM-Onto construction is the definition of data properties as it is shown in Table III. The table presents also their ranges, their domains and their descriptions. Differently to object properties' ranges which are in form of ontology classes, the data properties ranges are data types. In other terms, they are either strings of character (string) or real numbers (float) or entire numbers (Int).

*B. Product Semantic Model Generation*

After the construction of the ADM-Onto, the suitable semantic model is generated through the second module of the proposed system.

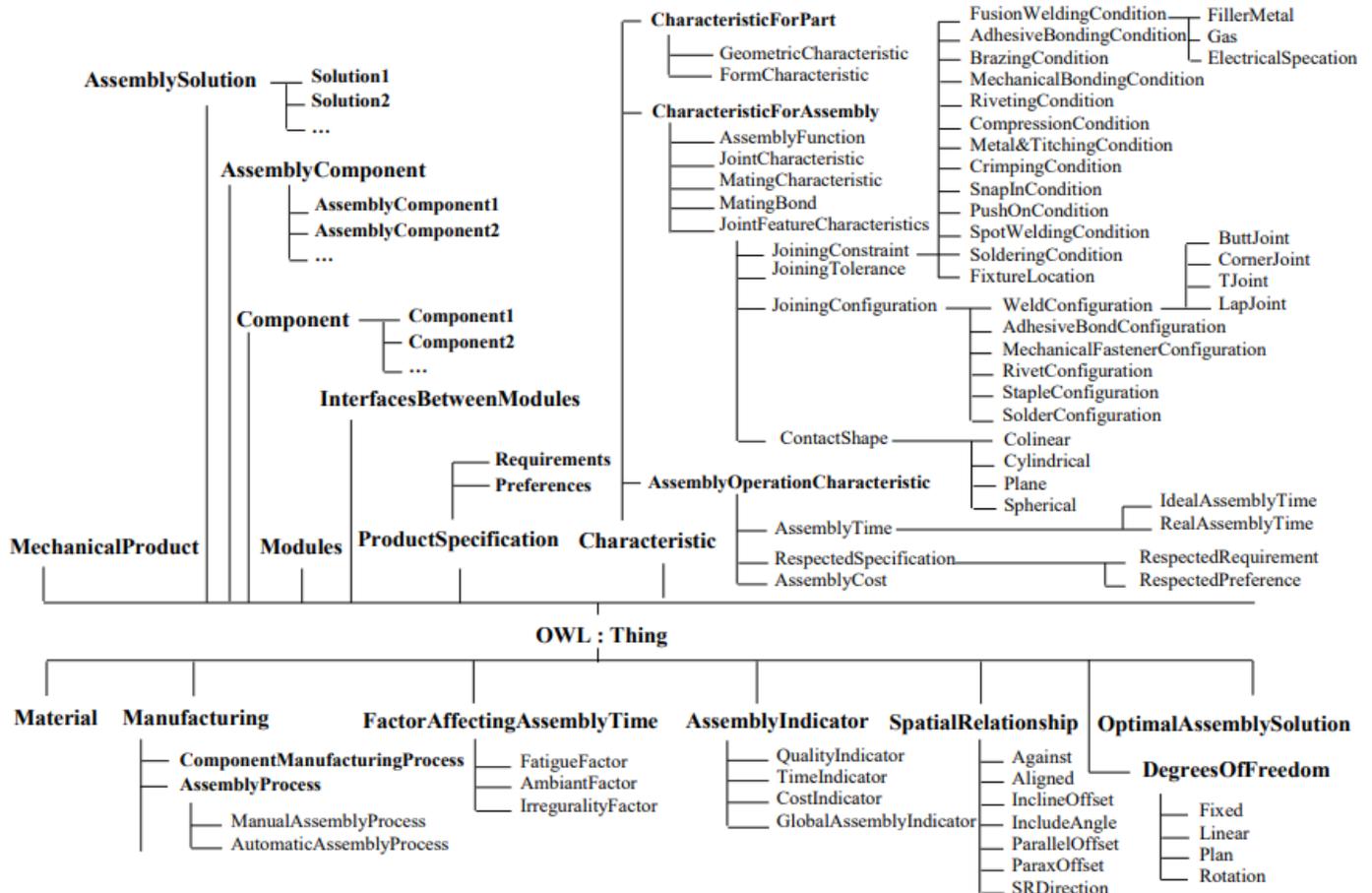


Fig. 3. Classes of the ADM-Onto.

TABLE II. OBJECT PROPERTIES OF THE ADM-ONTO

No.	Object Property	Domains	Ranges	Inverse Property
1	HasPart	MechanicalProduct	Modules InterfacesBetweenModules	IsPartOf
2	ComposedOf	MechanicalProduct	Modules InterfacesBetweenModules	IsModuleOf
3	HasComponent	Modules	Component AssemblySolution	IsComponentOf
4	HasAssemblyComponent	AssemblySolution	AssemblyComponent	IsAssemblyComponentOf
5	IsAssembledWith	Component	"another" Component	IsAssembledWith
6	IsAssemblyAlternative	AssemblySolution	MechanicalProduct	CanBeAssembledWith
7	HasSpecification	MechanicalProduct	ProductSpecification	IsSpecificationOf
8	HasRequirement	MechanicalProduct	Requirement	IsRespectedBy
9	HasPreference	MechanicalProduct	Preference	
10	BelongTo	FormCharacteristic	Component	HasFormCharacteristic
11	BelongToFormCharacteristic	GeometricCharacteristic	FormCharacteristic	HasGeometriCharacteristic
12	HasMaterial	Component AssemblyComponent	Material	IsMadeWith
13	HasReferenceDirection	MatingCharacteristic	SRDirection	IsDirectionOf
14	HasMatingComponent	MatingCharacteristic	FormCharacteristic	*****
15	HasProcess	MechanicalProduct	Manufacturing	IsProcessOf
16	HasComponentProcess	Component	ComponentManufacturingProcess	IsCompnentProcessOf
17	HasAssemblyProcess	Solution	AssemblyProcess	IsAssemblyProcessOf
18	IsDoneIn	Manufacturing	AssemblyTime	IsTimeOf
19	IsDoneInIdealTime	AssemblyProcess	IdealAssemblyTime	IsIdealAssemblyTimeOf
20	IsDoneRealTime		RealAssemblyTime	IsRealAssemblyTimeOf
21	HasRespectedSpecification	AssemblySolution	Specification	IsRespectedBy
22	HasRespectedRequirement	AssemblySolution	Requirement	
23	HasRespectedPreference	AssemblySolution	Preference	
24	HasFactorAffectingAssemblyTime	AssemblyProcess AssemblySolution	FactorAffectingAssemblyTime	IsFactorAffectingAssemblyTime
25	HasAssemblyFatigueFactor	AssemblyProcess AssemblySolution	FatigueFactor	IsFatigueFactorFor
26	HasAssemblyAmbiantFactor	AssemblyProcess AssemblySolution	AmbliantFactor	IsAmbiantFactorFor
27	HasAssemblyIrregularityFactor	AssemblyProcess AssemblySolution	IrregularityFactor	IsIrregularityFactorFor
28	HasAssemblyIndicator	AssemblyProcess AssemblySolution	AssemblyIndicator	IsAssemblyIndicatorFor
29	HasAssemblyQualityIndicator	AssemblyProcess AssemblySolution	QualityIndicator	IsQualityIndicatorFor
30	HasAssemblyTimeIndicator	AssemblyProcess AssemblySolution	TimeIndicator	IsTimeIndicatorFor
31	HasAssemblyCostIndicator	AssemblyProcess AssemblySolution	CostIndicator	IsQualityIndicatorFor
32	HasGlobalIndicator	AssemblyProcess AssemblySolution	GlobalAssemblyIndicator	IsGlobalAssemblyIndicatorFor
33	HasOptimalSolution	MechanicalProduct	OptimalAssemblySolution	IsOptimalAssemblySolutionFor

TABLE III. DATA PROPERTIES OF THE ADM-ONTO

Data Property	Domains	Range	Description
HasTechnicalFunction	MechanicalProduct Component AssemblyComponent Module	String	To indicate the technical function of the studied mechanical product/ a module/ a component/ an assembly component
IsRepeated	Module Component AssemblyComponent	Int	To record how many times the module/ the component/the assembly component is repeated in the studied mechanical product. So that, the redundancy of analysis is avoided
HasWeight	MechanicalProduct Component AssemblyComponent Module	Float	To record the weight of the studied mechanical product/ a module/ a component/ an assembly component
HasModel	MechanicalProduct Component AssemblyComponent Module	String	To record the model of the studied mechanical product/ a module/ a component/ an assembly component
IsHazardous	Material	Boolean	To indicate whether a material is hazardous
IsAssembledInPlusX	Module AssemblyComponent Component	Boolean	To record the assembly-direction(s) of a module/ an assembly component/ a component
IsAssembledInMinusX			
IsAssembledInPlusY			
IsAssembledInMinusY			
IsAssembledInPlusZ			
IsAssembledInMinusZ			
HasSimilarityMemoryIndex	MechanicalProduct	Float	This property is used to realize the case updating strategy
IsManualProcess	AssemblyProcess	Boolean	To indicate if the assembly process is manual or automatic
IsAutomaticProcess	AssemblyProcess	Boolean	
HasNbrOfRespectedRequirement	Solution	Int	To record the number of requirement respected by the solution
HasNbrOfRequirement	MechanicalProduct	Int	To record the number of requirement imposed by costumers
HasState	Solution	Boolean	This property is used in the RBR process
HasNbrOfRespectedSpecification	Solution	Int	To record the number of specifications respected by the solution
HasNbrOfSpecificaion	MechanicalProduct	Int	To record the number of specifications imposed by the design team at the beginning of the design study
HasQualityIndicator	Solution	Float	This property is used to calculate the quality indicator of the solution
HasFatigueFactor	Solution	Float	To record the fatigue factor of the solution
HasAmbiantFactor	Solution	Float	To record the ambient factor of the solution
HasIrregularityFactor	Solution	Float	To record the irregularity factor of the solution
HasIdealTime	Solution	Float	To record the ideal time to realize the solution
HasRealTime	Solution	Float	To record the real time to realize the solution
HasTimeIndicator	Solution	Float	This property is used to calculate the time indicator of the solution
HasTotalAssemblyCost	Solution	Float	To record the total assembly cost of a solution
HasTotalAverageCostOf Production	MechanicalProduct	Float	To record the total average cost of production of the studied mechanical product imposed by the design team at the beginning of the design study
HasCostIndicator	Solution	Float	This property is used to calculate the cost indicator of the solution
HasGlobalAssemblyIndicator	Solution	Float	This property is used to calculate the global indicator of the solution
HasHighestGlobalIndicator	Solution	Boolean	this property is used to indicate the solution, that has the highest global indicator, which is the optimal one

The generated semantic model of an assembly case is presented in Fig. 4. It is compound of two parts:

- The semantic model of the studied mechanical product:

This first part describes the structure of the product (its architecture). In fact, through the "HasPart" object property, relations between the different product components are defined. This relations definition is done at two levels: the "MechanicalProduct/Module" level that records the different modules which compose the Mechanical Product; and the "Module/Component" level which records the different components that compose each module. By its turn, each product component can be decomposed to different subcomponents.

- The semantic model of the optimal assembly solution:

This second part is the continuity of the first part (The semantic model of the studied mechanical product) work.

In this part, the "IsAssembledWith" object property defines two types of relations: Module/Module relation which record modules that should be assembled together; and

Component/component relation which record components that should be assembled with each other and their natures. The "InterfacesBetweenModules" class plays also a key role in the definition of the first relation (Module/Module) in terms of types and numbers of links between the studied product modules. Brief, this first part of the optimal assembly solution semantic model allows the definition of the different connections' types between modules and components.

For the second part of the optimal assembly solution semantic model, the goal is to define the different assembly alternatives of the studied mechanical product. This purpose is achieved through the "IsAssemblyAlternative" object property and the "HasGlobalAssemblyIndicator" data property. It is in this part that the different assembly alternatives are ranked.

Then, by the combination of those two parts, the optimal assembly solution is obtained. This result is got through also the help of the "HasOptimalSolution" object property and the "HasHighestGlobalIndicator" data property that affects the optimal solution  $S^*$  to the one that has the highest global indicator value.

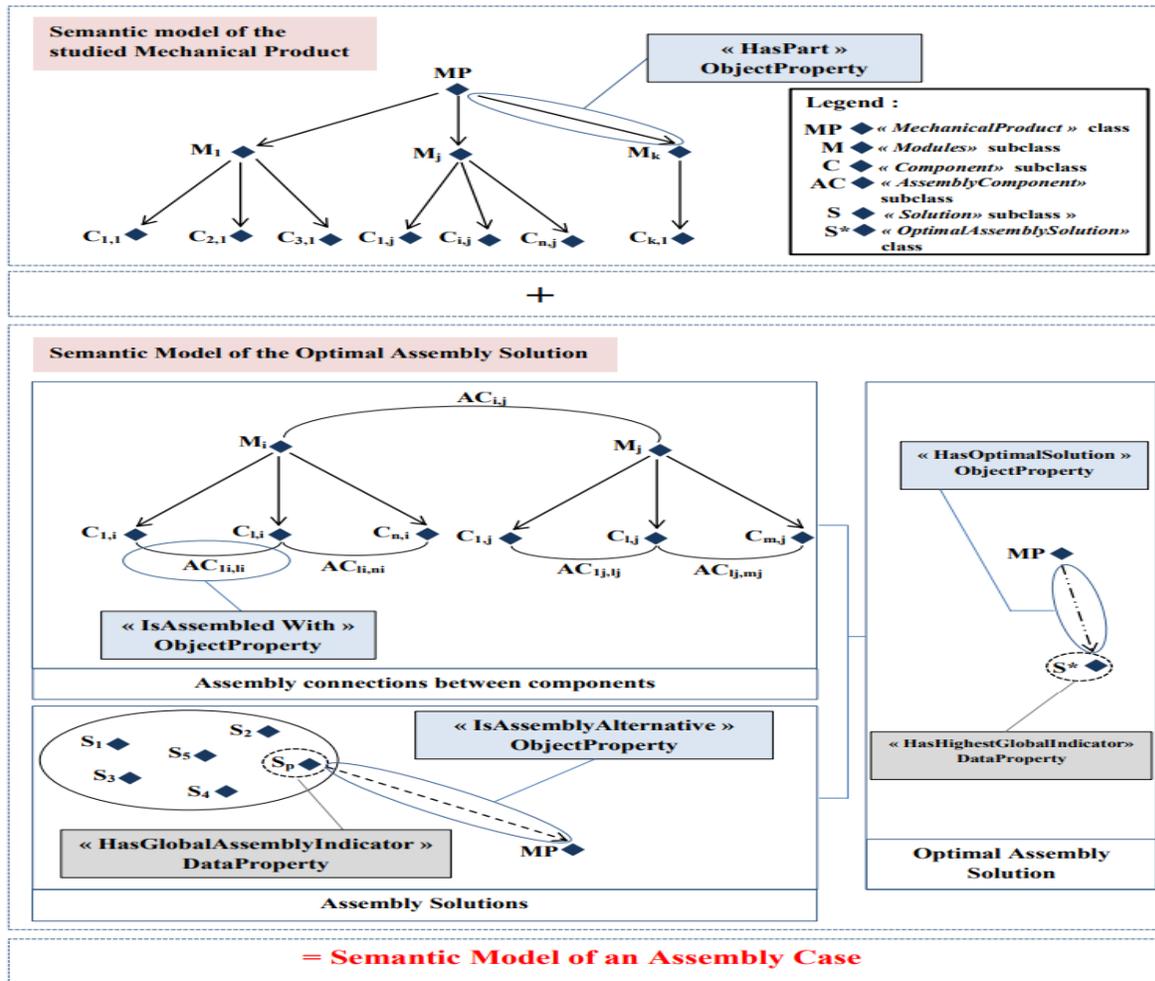


Fig. 4. Semantic Model of an Assembly Case.

## V. DESCRIPTION OF THE CASE-BASED REASONING (CBR) MODULE FOR OPTIMAL ASSEMBLY SOLUTION SELECTION

The CBR module aims to use the previous cases of assembly solution selection studies in a new case. Thus, this module is based on a base case that contains the archive of studies done previously by the design team.

In this section, the adopted notation is as follow:

- $b$  : number of case studies belonging to the case base
- $CS_t$  : Case Study NO  $t$  belonging to the case base with  $t=1, \dots, b$
- $k$  : number of modules of the studied mechanical product
- $M_j$  : Module number  $j$ ,  $j=1, \dots, k$
- $m_j$  : Total number of  $M_j$  components
- $C_{ij}$  : The product component number  $i$  of module  $j$  with  $i=1, \dots, m_j$  and  $j=1, \dots, k$
- $l_j$  : Number of the similar modules to  $M_j$
- $M_{j,sl}$  : The similar module number  $l$  to  $M_j$  with  $l=1, \dots, l_j$
- $n_{jl}$  : Number of  $M_j$  component that have similar one in  $M_{j,sl}$
- $C_{ij,st}$  : A component  $C_{ij}$  that has similar one in the case study  $C_{st}$
- $C_{ij,ns,d}$  : A component  $C_{ij}$  that has no similar one in the case base number  $d$
- $AC_{il,j}$  : Assembly connection between  $C_{ij}$  and  $C_{il}$
- $r_j$  : total number of connections in  $M_j$
- $AC_{il,j,st}$  : A similar assembly connection to  $AC_{il,j}$  in  $CS_t$
- $e_{jl}$  : Number of  $M_j$  assembly connections that have similar one in  $M_{j,sl}$
- $AC_{il,j,np}$  : An assembly connection  $AC_{il,j}$  that has no similar one in  $CS_p$
- $M_j^*$  : The most similar case to  $M_j$
- $S_j^*$  : The set of the similar components and connections in  $M_j$  to the ones in  $M_j^*$
- $NS_j^*$  : The set of the remaining components and connections in  $M_j$  ( the ones that not have similarities in  $M_j^*$  )

Fig. 5 presents the proposed working process of the CBR module.

The first step in the CBR process is to search, in the base case, the most similar cases to the studied one. To do this, the research will be done per module. In fact, for each module  $M_j$ , an analysis is done in the ADM-Onto memory to check if similar modules, that are composed of similar components and

connections and that can ensure the same functions with some minor modifications, have been previously developed by the design team. This operation is done by considering firstly the previous complete cases/studies as an individual of the *MechanicalProduct* class. Then, designers have to use three different elements, in order to determine cases that should be compared to the studied module  $M_j$ , namely:

- The Module class name as an index
- The "HasPart", the "IsAssembledWith", the "BelongTo", the "BelongToForm", the "HasMaterial", the "IsManufacturedUsing" and the "HasProcess" object properties
- The "HasTechnicalFunction" and the "HasModel" data properties.

By determining all the alternatives, the case-comparison is the next step in the CBR process. In this context, a Similarity Memory Index (SI) is proposed. It is based on two sides which are the similarities between  $M_j$  and  $M_{j,sl}$  modules in terms of similar components and similar connections.

It is to note that  $AC_{ik,j,sl}$  refers to the existence of a similar connected components to  $C_{ij}$  and  $C_{kj}$  in  $M_{j,sl}$  in terms of connections. In other terms,  $C_{ij}$  and  $C_{kj}$  should have the same technical functions to their similar components and the same degrees of freedom to remove/ to block in order to ensure the assembly solution.

So for each alternative  $M_{j,sl}$ , SI is computed using the following formula:

$$SI_{j,l} = \frac{1}{2} * \left( \frac{n_{jl}}{m_j} + \frac{e_{jl}}{r_j} \right) \quad (2)$$

The most similar module to the studied one  $M_j$  is the one that has the higher SI value:  $M_{j,s}^*$ .

By determining  $M_{j,s}^*$ , Two cases are supposed to have:

- $SI=1$ : which means that  $M_j$  and  $M_j^*$  are totally similar. In this case, designers should reuse and extract the optimal assembly solution from  $M_j^*$ .
- $0 < SI < 1$ : which means that  $M_j$  and  $M_j^*$  are not totally similar. In this case, designers should extract the optimal assembly solution from  $M_j^*$  for the set of the similar components and connections  $S_j^*$ . Then, they apply the RBR method to determine the optimal assembly solution for the set of the remaining components and connections  $NS_j^*$ .

The execution of the CBR decision-making process for optimal assembly solution selection is based on a main principal: the ADM-Onto can express cases in different granularity. In fact, a module of the *MechanicalProduct* class can be a complete previous product/case (which contains in its turn multiple components/ sub-assemblies) and each two assembled individuals has a corresponding optimal assembly solution.

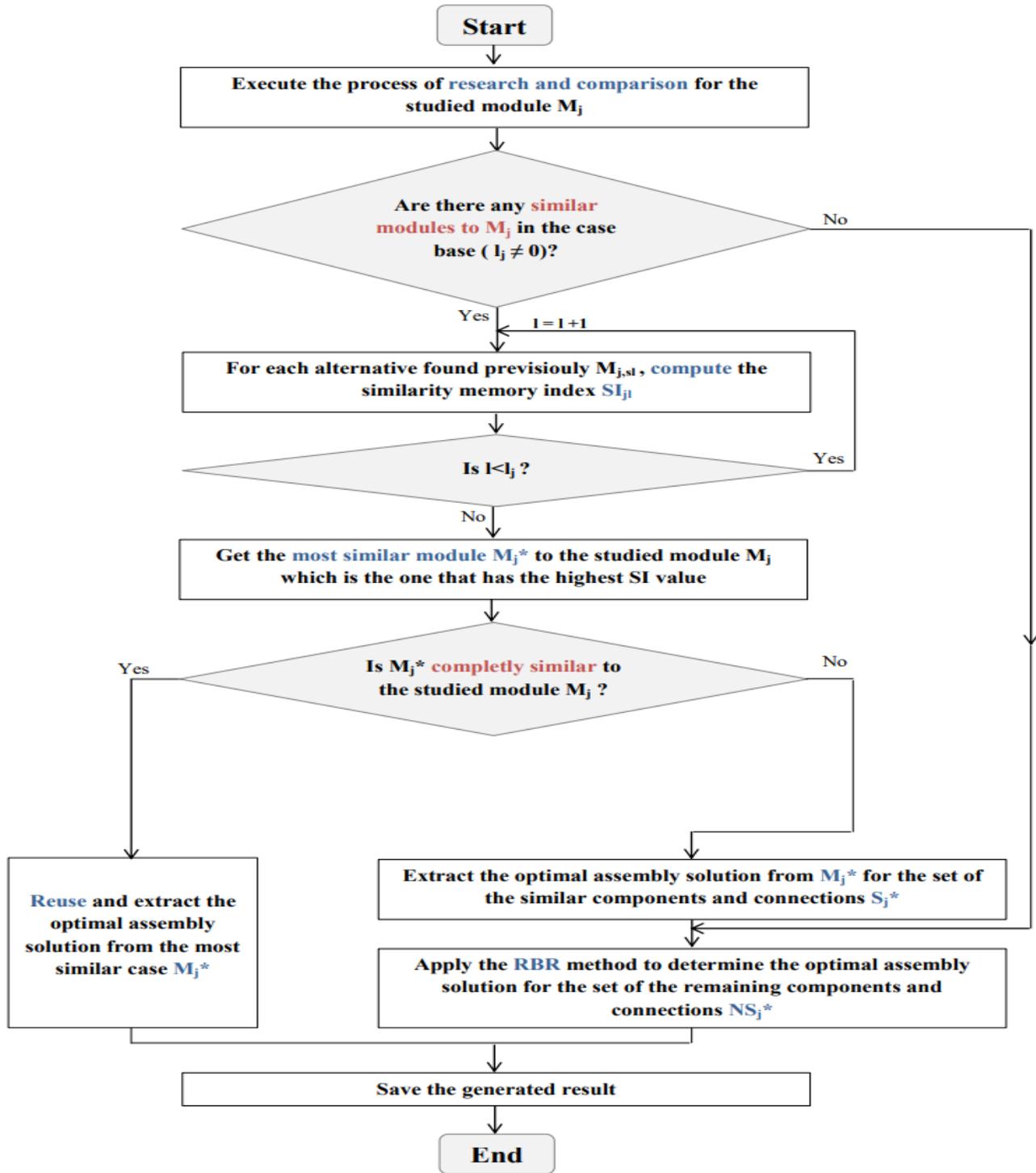


Fig. 5. The Proposed CBR based Decision-Making Process for Optimal Assembly Solution Selection.

VI. DESCRIPTION OF THE RULE-BASED REASONING (RBR) MODULE FOR OPTIMAL ASSEMBLY SOLUTION SELECTION

The use of the RBR module is the second solution to select the optimal assembly alternative if the CBR module does not succeed to do it. In fact, as its name indicates, the RBR module is based on a rules base that designers should define and execute on the ADM-Onto in order to obtain automatically the optimal assembly solution of the studied mechanical product.

In the proposed RBR module, the integrated DFMMA, in particular its proposed assembly solution selection methodology is the base of the defined rules as shown in Table IV.

In addition to that, those rules are expressed using SWRL/SQWRL [33-35]. Thus, the listed rules are in the form of implication between an antecedent and its consequence.

So, if antecedent conditions hold (are "True"), then the consequent conditions must also hold. In SWRL, rules parts, i.e. their antecedent and their consequents, are formed basing on a set of axioms written in the following form: a1^.....^an. Each axiom refers to individuals, data literals, individual variables or data variables. All those variables are defined in the standard convention form that consists on prefixing them with a question mark (e.g. ?V).

The proposed working process of the RBR module is presented in Fig. 6.

For each module of the studied mechanical product, the RBR working process is executed. It passes by four steps:

- Elimination of alternatives that not satisfy costumers' requirement:

In effect, the execution of rules 1 and 2 classifies the different assembly alternatives of the studied module in function of their satisfaction of costumers' requirements. The result of this two rules is given by affecting the "0" value to the data property "HasState" for alternatives that not satisfy them and the "1" value for the other ones. Then, to eliminate

the first category of alternatives, the rule 3 is executed. It aims to affect the value "0" to their related "HasGlobalAssemblyIndicator" data property ranges. Thus, all those alternatives will have automatically the lowest global indicator value and will be eliminated.

- Computation of indicators for the remaining alternatives:

This step is succeeded through the execution of rules 4, 5, 6 and 7. In fact, it aims to compute in an automatic way the four assembly indicators for alternatives that have "HasState = 1" data property (i.e. that satisfies costumers requirements) notably the quality indicator, the time indicator, the cost indicator and finally the global assembly indicator.

Those indexes are the base of selection according to the integrated DFMMA approach, in particular the optimal assembly solution selection part.

- Rank of alternatives basing on their global assembly indicator:

This step is based on the results of the previous step. Actually, it aims to rank alternatives from the one that has the higher global assembly indicator to the one that has the lower one.

- Identification of the optimal alternative and save the generated result:

This step is the last one in the RBR process. It aims to relate the studied "module" class to the "optimalAssemblySolution" class through the "HasOptimalSolution" object property and the "HasHighestGlobalIndicator" data property. Then, the result of the study, namely the optimal assembly solution of the studied module, is saved in the case base of the ADM-Onto. This saved result can be used and can help the team design later to solve other new cases by executing the CBR process. Thus, this step plays a key role to save time and consequently money in the future design studies.

TABLE IV. ASSEMBLY EXECUTED RULES OF THE PROPOSED RBR MODULE

NO	Rules
1	AssemblySolution(?S) ^ HasNbrOfRespectedRequirement(?S, ?r) ^ MechanicalProduct(?P) ^ HasNbrOfRequirement(?P, ?n) ^ IsAssemblyAlternative(?S, ?P) ^ swrlb:lessThan(?r, ?n) ^ → HasState(?S, 0)
2	AssemblySolution(?S) ^ HasNbrOfRespectedRequirement(?S, ?r) ^ MechanicalProduct(?P) ^ HasNbrOfRequirement(?P, ?n) ^ IsAssemblyAlternative(?S, ?P) ^ swrlb:equal(?r, ?n) ^ → HasState(?S, 1)
3	AssemblySolution(?S) ^ HasState(?S, 0) → HasGlobalAssemblyIndicator(?S, 0)
4	AssemblySolution(?S) ^ HasState(?S, 1) ^ HasNbrOfRespectedSpecification(?S, ?N) ^ MechanicalProduct(?P) ^ HasNbrOfSpecification(?P, ?M) ^ swrlb:divide(?t, ?N, ?M) ^ swrlb:multiply(?Q, ?t, 100) → HasQualityIndicator(?S, ?Q) ^ sqwrl:select(?S, ?Q) ^ sqwrl:columnNames("Assembly Alternative", "Quality Indicator_in %")
5	AssemblySolution(?S) ^ HasState(?S, 1) ^ HasFatigueFactor(?S, ?Ffa) ^ HasAmbiantFactor(?S, ?Fam) ^ HasIrregularityFactor(?S, ?Fir) ^ hasIdealTime(?S, ?Tid) ^ HasRealTime(?S, ?Tre) ^ swrlb:multiply(?q, ?Ffa, ?Fam, ?Fir, ?Tid, 100) ^ swrlb:divide(?T, ?q, ?Tre) → HasTimeIndicator(?S, ?T) ^ sqwrl:select(?S, ?T) ^ sqwrl:columnNames("Assembly Alternative", "Time Indicator_in %")
6	AssemblySolution(?S) ^ HasState(?S, 1) ^ HasTotalAssemblyCost(?S, ?Cat) ^ HasTotalAverageCostOfProduction(?S, ?CavPro) ^ swrlb:divide(?T, ?Cat, ?CavPro) ^ swrlb:subtract(?q, 1, ?T) ^ swrlb:multiply(?C, ?q, 100) → HasCostIndicator(?S, ?C) ^ sqwrl:select(?S, ?C) ^ sqwrl:columnNames("Assembly Alternative", "Cost Indicator_in %")
7	AssemblySolution(?S) ^ HasState(?S, 1) ^ HasQualityIndicator(?S, ?Iq) ^ HasTimeIndicator(?S, ?It) ^ HasCostIndicator(?S, ?Ic) ^ swrlb:multiply(?a, ?Iq, ?It, ?Ic) ^ swrlb:divide(?G, ?a, 10000) → HasGlobalAssemblyIndicator(?S, ?G) ^ sqwrl:select(?S, ?G) ^ sqwrl:columnNames("Assembly Alternative", "Global Assembly Indicator_in %")

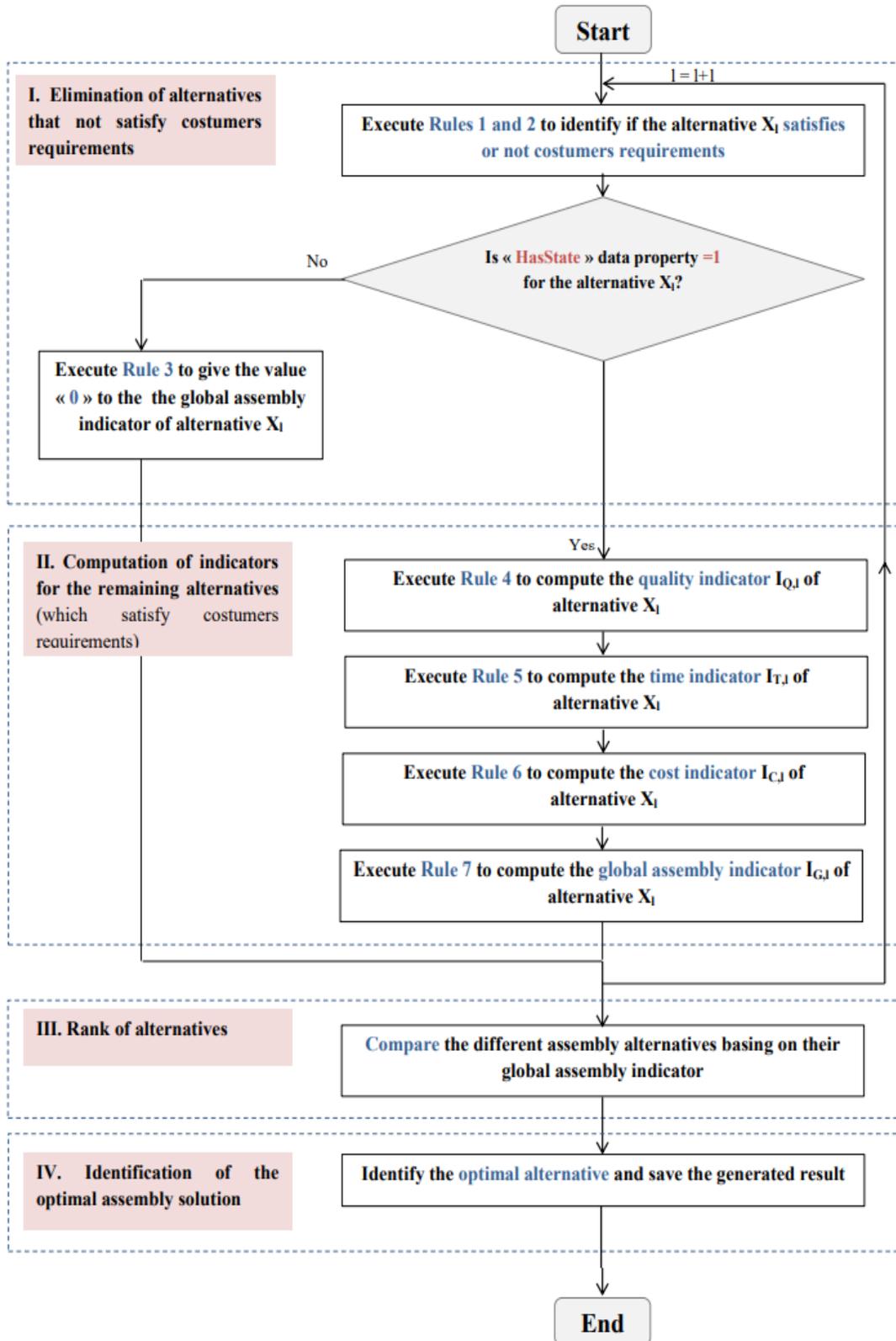


Fig. 6. The Proposed RBR based Decision-Making Process for Optimal Assembly Solution Selection for each Module of the Studied Mechanical Product.

## VII. CASE OF STUDY

To illustrate the functioning and the applicability of the proposed ADM system, a case of study is presented in this section. Actually, the different steps of the proposed methodology are applied in this section on a complex mechanical product which is the Schrader robot.

The main function of this studied product is moving objects from one station to another. To ensure this function, different modules are assembled together, namely a rotating base, an elevation unit, a linear unit, a rotating wristband and finally a grip. In addition to that, according to the integrated DFMMA approach, those modules are linked with mechanical interfaces as it is shown in Fig. 7.

By defining the Schrader robot architecture, its modules and interfaces between them, its components materials and their manufacturing processes, according to the DFMMA approach [1], all the useful information to select the optimal assembly solution are collected. Thus, the first step of the ADM methodology working process is implemented.

Then, the second step, which is the construction of the ADM-Onto, is applied. Thus, all its classes, its object properties and its data properties, in addition to all their domains and their ranges are defined.

As there are no similar cases in regards to all the Schrader robot modules in the case base of the assembly ontology, the CBR process cannot be used in this case. Thus, according to the proposed methodology, the RBR process is applied to select the optimal assembly solution for all modules of the studied mechanical product.

As an example, in this paper, the grip module, which is the most important module of the Schrader robot, will be the subject of the RBR process application. The function of this module is to Grip or put down the object to be handled.

The grip module is composed of six components which are: The body, a piston, two fingers and two connecting rods. All the details of this module functioning are presented in reference 1.

In this case of study, the design team has to compare three different assembly solutions. Actually, designers have choose to assembly the body and the piston by four screw nuts; Then, to assembly the piston to the two rods with a pin clamped by both extremities of the Schrader grip body.

The unique difference between the three alternatives consists on the assembly solution between the rods and the fingers. But, to keep the balance during the functioning of the studied product in terms of mechanical efforts and also in terms of the opening and closing speed of its fingers, the same solution is used for the two sides (the lower and the upper). Thus, the three assembly alternatives between the rods and the fingers components are as following:

- Assembly by pins + circlips
- Assembly by pins + washers
- Assembly by pins + locknuts

Before executing the different previous defined rules of the RBR process, the different assembly alternatives of the grip module are defined in the constructed ADM-Onto ontology.

It is to note that the used reasoner in the constructed ADM-Onto is "Pellet". This choice is done basing firstly on the different features that presents this reasoner.

In fact, it is characterized by its consistency checking. Consequently, the constructed ontology does not contain any contradictory facts. Furthermore, the concept of satisfiability is realized because of the ability of "Pellet" reasoner to check the possibility for a class to have any instances. Another feature of this used reasoner is its capacity of classification by computing the subclass relations between every named class to create the complete class hierarchy. It is able also to compute the direct types for each of the individuals.

In addition to all those features, Pellet reasoner presents a main advantage that makes it particular. Actually, differently to the other reasoners that detect inconsistent but the diagnosis and resolution of the bug is not supported at all, Pellet contains two additional debugging services. They help the user to know the inconsistency reasons:

- The service clash detection that permits the determination of the root contradiction or clash in the completion graph;
- The axiom tracing that allows the extraction of the relevant source axioms from the ontology responsible for the clash.

As previously explained, the first stage of the RBR process is to eliminate assembly solutions that not satisfy costumers' requirements. To do so, the first defined rules are executed firstly. The results are presented in Fig. 8.

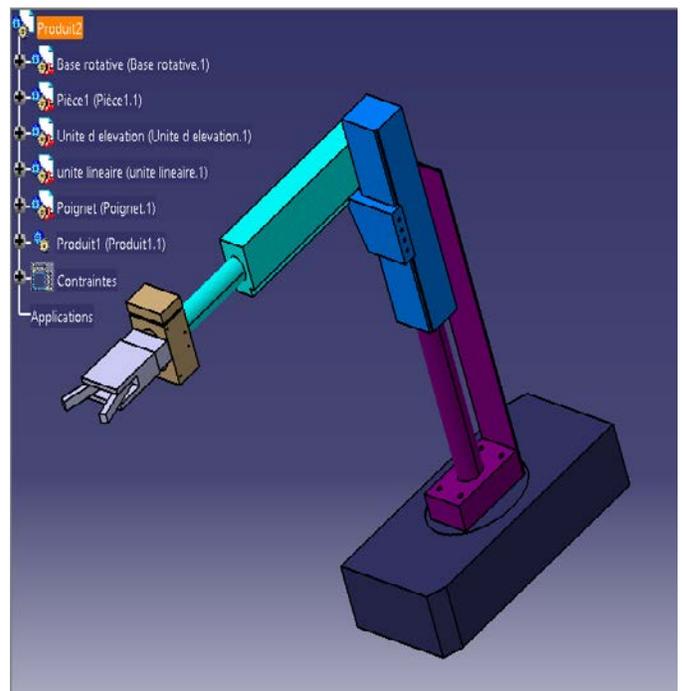


Fig. 7. The Schrader Robot Architecture: Its Different Modules.

All of the grip assembly alternatives have "HasState =1" data property. Consequently, any of them is eliminated by the execution of Rule 3.

The next stage is to compute the different indexes defined according to the integrated DFMMA approach, in particular its assembly solution selection methodology, by executing rules 4, 5, 6 and 7. Fig. 9, Fig. 10, Fig. 11 shows respectively

the different obtained values of the quality indicators, the time indicators and the cost indicators with the reasoner Pellet.

Fig. 12 presents the global assembly indicator results for all the compared grip assembly alternatives.

According to those results, designers conclude that the optimal assembly alternative is the first one since it has the highest global index (Fig. 13).

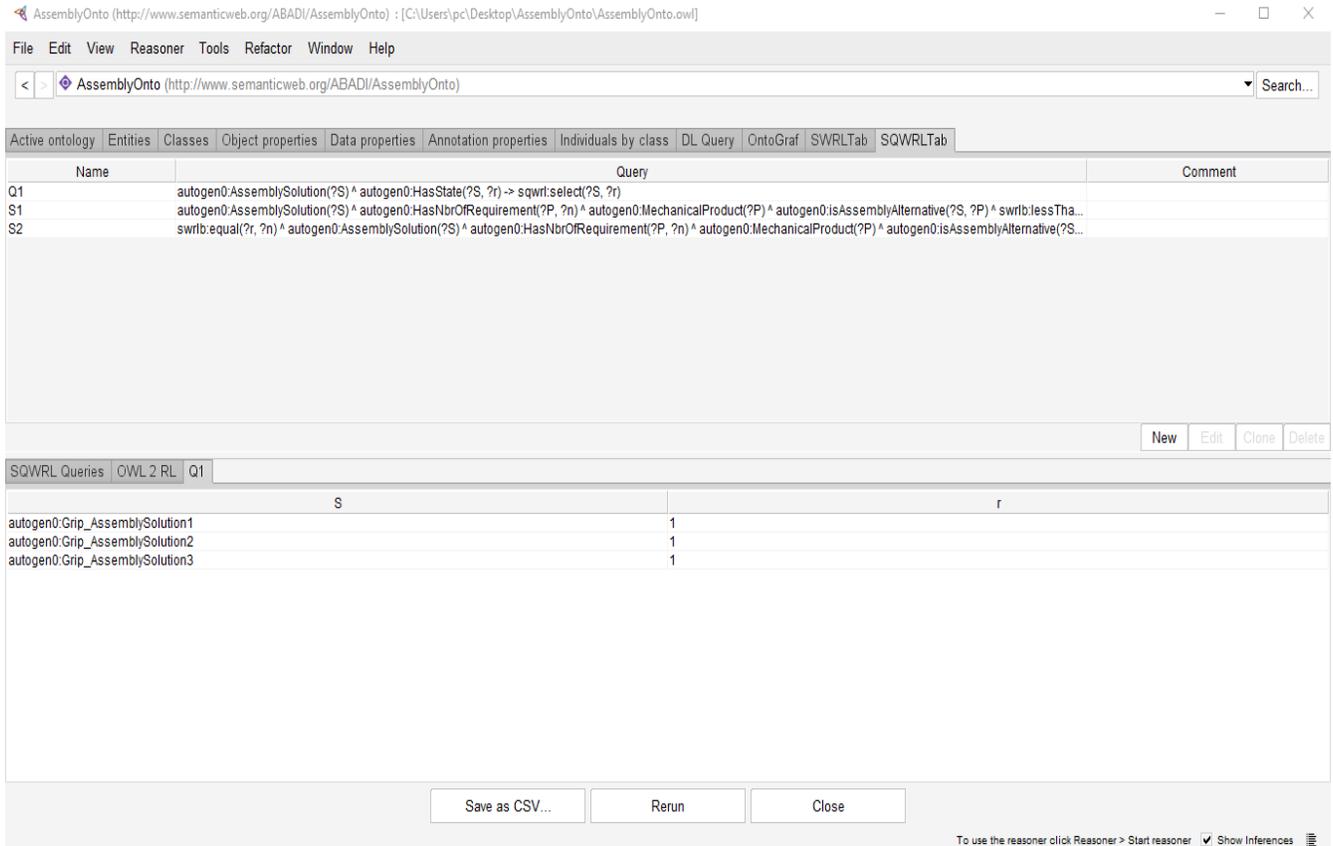


Fig. 8. Results of the First Phase of the RBR Process Application.

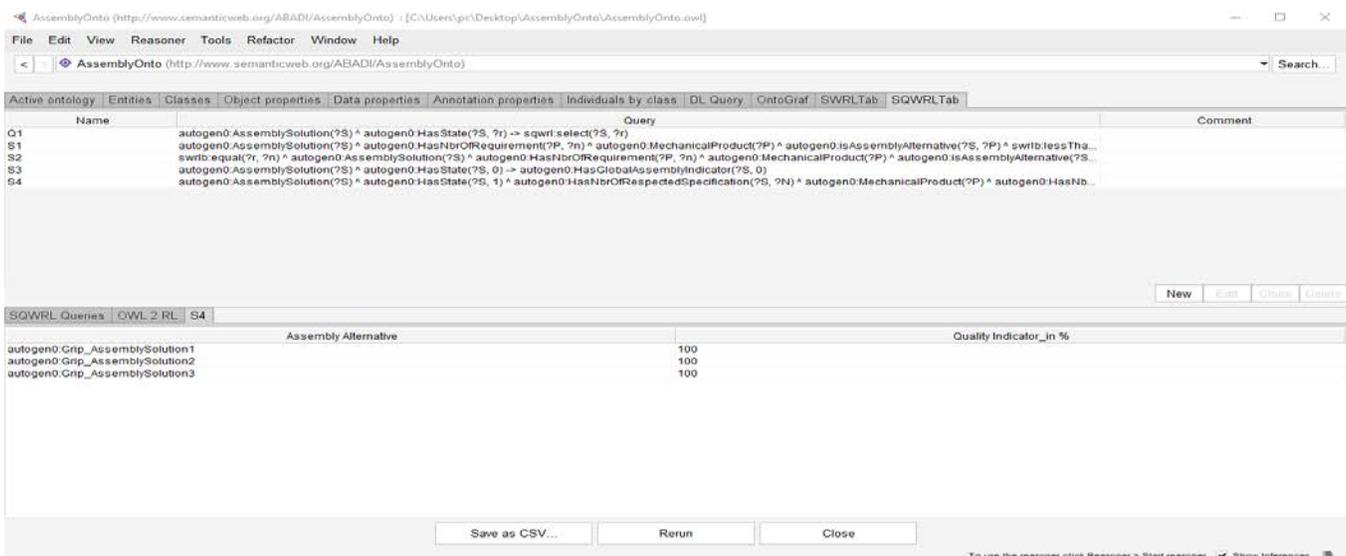


Fig. 9. Quality Indicators of the different Grip Assembly Alternatives.

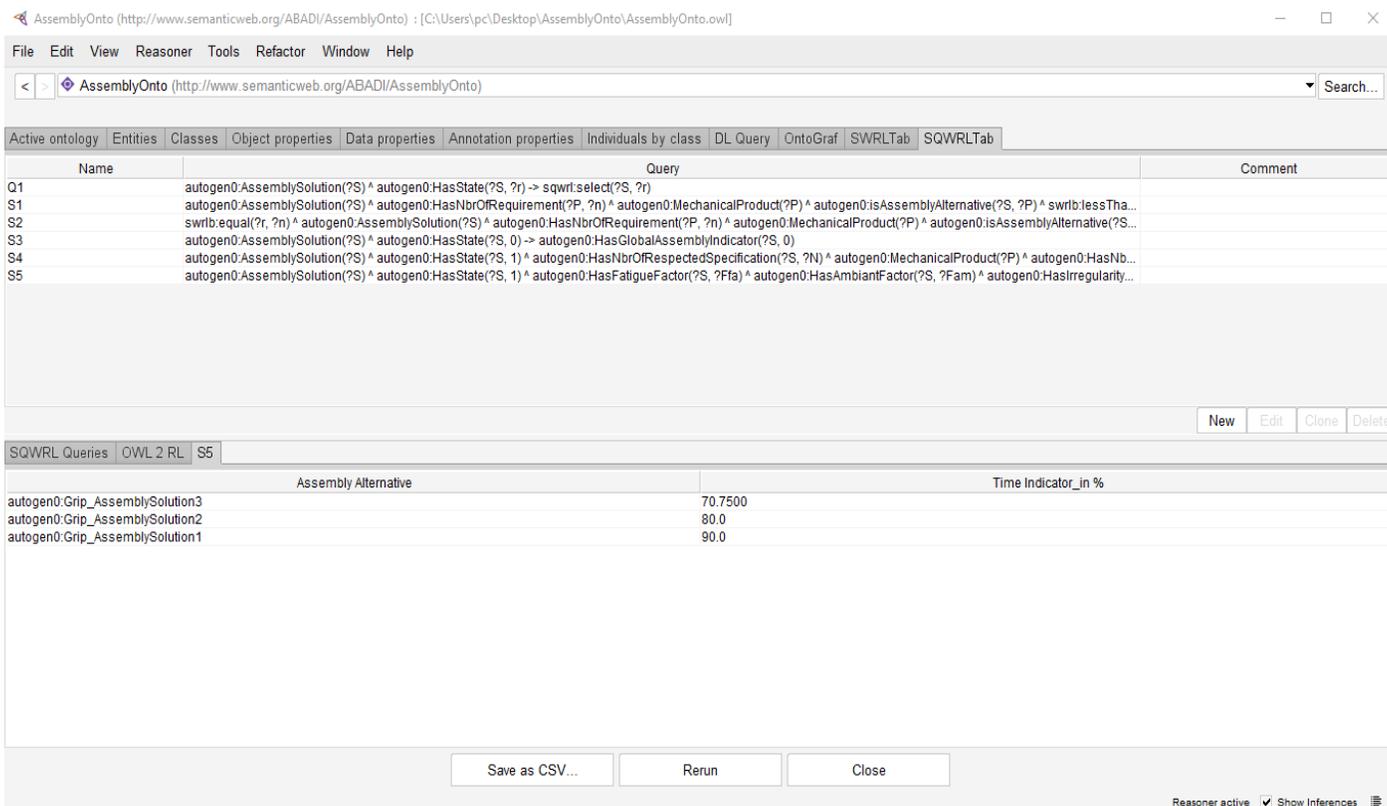


Fig. 10. Time Indicators of the Different Grip Assembly Alternatives.

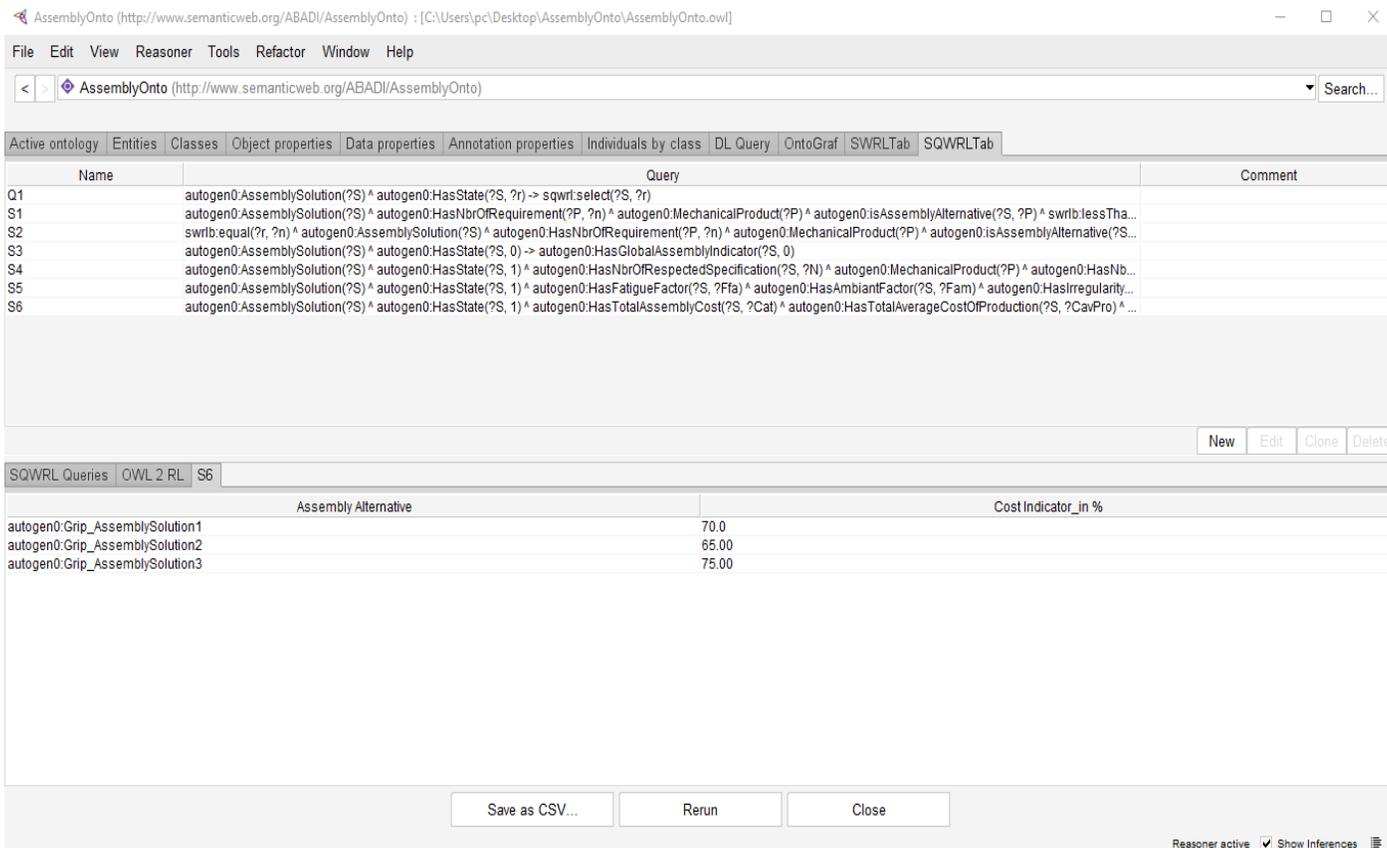


Fig. 11. Cost Indicators of the Different Grip Assembly Alternatives.

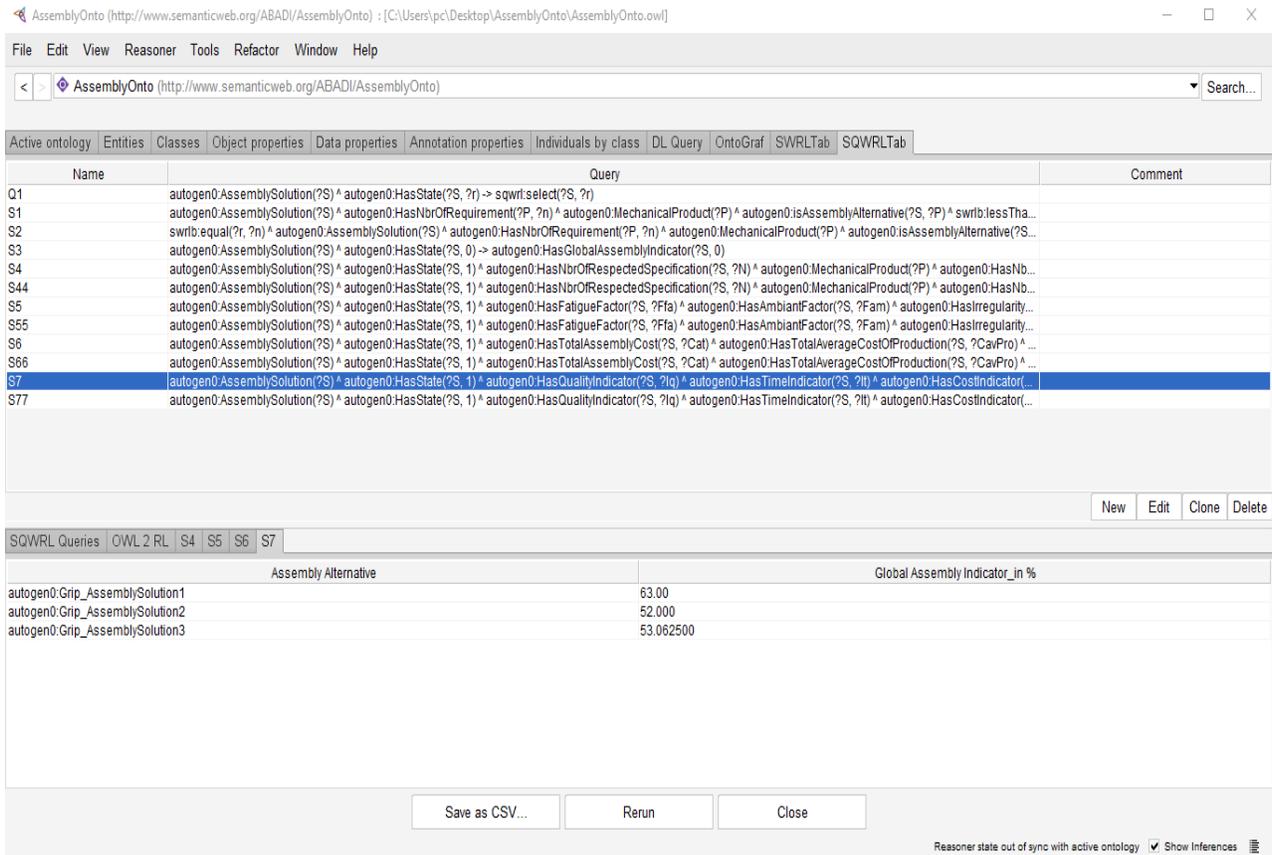


Fig. 12. Global Assembly Indicators of the Different Grip Assembly Alternatives.

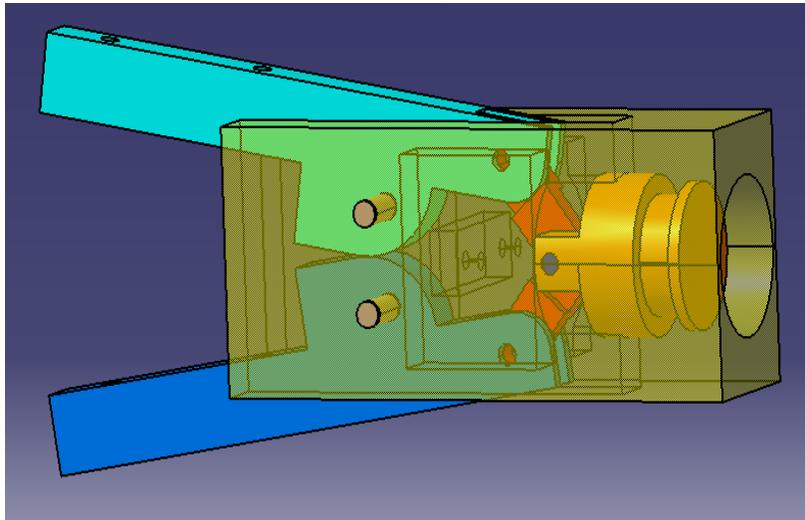


Fig. 13. The Optimal Assembly Solution of the Grip Module.

### VIII. CONCLUSION AND PERSPECTIVES

In this paper, a flexible and automated decision making system is proposed. The developed ADM system is composed of several modules and it aims to automate the selection of the optimal assembly solution. It is based on ontologies, CBR and RBR concepts. In addition to that, the proposed methodology/system is an automation of the integrated

DFMMA approach, in particular its assembly solution selection methodology.

Unlike the manual assembly selection previous works, the proposed system uses the previous developed cases of the design team in the new ones and benefits from its experience basing on the Case Based Reasoning. In addition to that, through the Rule Based Reasoning used in the developed

system, the selection is automatic and easy even if the number of assembly alternatives is high.

To validate all the advantages of our ADM system, a case of study is presented in the end. Actually, the ADM methodology is applied on a mechanical complex product: The Schrader Robot. The implementation of the proposed automated methodology on the considered case study permits the definition of its different modules, components and interfaces between modules in the ADM-Onto in forms of classes. In addition to that, it allows the definition of the different relations between them through different object and data properties. Finally, the selection of the optimal assembly solution is done automatically by the execution of the different rules of the ADM system RBR module.

As perspectives, it is suggested to enrich more the proposed ADM-system, in particular its ontology ADM-Onto, by integrating other aspects to the assembly solution selection, namely, the security side of assembly operations and the environmental constraints. Another perspective is to consider uncertainties that presents the design phase in the future works.

#### REFERENCES

- [1] Abadi, A. Abadi, I. Massouri, An integrated product- process design approach considering material selection and product assembly. In: Journal of Advanced Manufacturing Systems, 2019 DOI: 10.1142/S0219686720500389, In press.
- [2] W. Beitz, G. Pahl, & K. Grote, Engineering design: a systematic approach. MRS BULLETIN, . 1996, pp. 71
- [3] T. Suzuki, T. Ohashi, M. Asano, & S. Miyakawa, Assembly reliability evaluation method (AREM). 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), May 2001, pp. 294-299. IEEE.
- [4] Lucas Engineering Systems Ltd. Design For Manufacture and Assembly Practitioners Manual, 1993, Version 10.
- [5] G. Boothroyd, Product design for manufacture and assembly. Computer-Aided Design, vol. 26(7), 1994, pp. 505-520.
- [6] S.N. Samy & H. ElMaraghy, A model for measuring products assembly complexity. International Journal of Computer Integrated Manufacturing, vol. 23(11), 2010, pp. 1015-1027.
- [7] S.N. Samy & K. H. ElMaraghy, Assessment of complexity measures suitable for mechanical assemblies. In Proceedings on 2nd CIRP Conference on Assembly Technologies and Systems, Toronto, ON, Canada, 2008.
- [8] R. B. Stone, K.L. Wood and R. H. Crawford, A heuristic method for identifying modules for product architectures. Des. Stud. vol. 21, 2000, pp. 5– 31.
- [9] C. Favi & M. Germani, A method to optimize assemblability of industrial product in early design phase: from product architecture to assembly sequence. International Journal on Interactive Design and Manufacturing (IJIDeM), vol. 6(3), 2012, pp. 155-169.
- [10] C. Favi, M. Germani & M. Mandolini, Development of complex products and production strategies using a multi-objective conceptual design approach. The International Journal of Advanced Manufacturing Technology, vol. 95(1-4), 2018, pp. 1281-1291.
- [11] G. B. Murali, B.B.V. L Deepak, B.B. Biswal & B. K. Khamari, Integrated design for assembly approach using ant colony optimization algorithm for optimal assembly sequence planning. In Computational intelligence in data mining, 2019, pp. 249-259. Springer.
- [12] S. Staab & R Studer (Eds.), Handbook on ontologies. Springer Science & Business Media, 2010.
- [13] C. Abadi, I. Manssouri, A. Abadi, A fuzzy ontology based approach to support product eco-design. A2IA. Springer, 2020, In press.
- [14] V. Fortineau, Contribution à une modélisation ontologique des informations tout au long du cycle de vie du produit (Doctoral dissertation). 2013.
- [15] F. A. Norki, R. Mohamad & N. Ibrahim, Context Ontology in Mobile Applications. Journal of ICT, vol. 19(1), 2020, pp. 21-44.
- [16] Y. B. Kang, S. Krishnaswamy, W. Sawangphol, L. Gao & Y. F. Li, Understanding and improving ontology reasoning efficiency through learning and ranking. Information Systems, vol. 87, 2020, 101412.
- [17] W. Li, H. Kang, D. Ma & W. Wei, SWRL Parallel Reasoning Implementation with Spark SQL. In IOP Conference Series: Materials Science and Engineering, vol. 719, No. 1, 2020, p. 012020. IOP Publishing.
- [18] T. Grubic & I. S. Fan, Supply chain ontology: Review, analysis and synthesis. Computers in Industry, vol. 61(8), 2010, pp. 776-786.
- [19] S. Cao, K. Bryceson & D. Hine, Improving supply chain risk visibility and communication with a multi-view risk ontology. In Supply Chain Forum: An International Journal, vol. 21, No. 1, 2020, pp. 1-15. Taylor & Francis.
- [20] C. E. Catalano, E. Camossi, R. Ferrandes, V. Cheutet & N. Sevilimis, A product design ontology for enhancing shape processing in design workflows. Journal of Intelligent Manufacturing, vol. 20(5), 2009, pp. 553-567.
- [21] A. Talhi, V. Fortineau, J.C. Huet & S. Lamouri, Ontology for cloud manufacturing based product lifecycle management. Journal of Intelligent Manufacturing, vol. 30(5), 2019, pp. 2171-2192.
- [22] M. M. Ali, M.B. Doumbouya, T. Louge, R. Rai & M. H. Karray, Ontology-based approach to extract product's design features from online customers' reviews. Computers in Industry, 2020, pp. 116, 103175.
- [23] J. Rockwell, I. R. Grosse, S. Krishnamurty & J. C. Wileden, A Decision Support Ontology for collaborative decision making in engineering design. In 2009 International Symposium on Collaborative Technologies and Systems, 2009, pp. 1-9. IEEE.
- [24] W.Y. Zhang & J. W. Yin, Exploring Semantic Web technologies for ontology-based modeling in collaborative engineering design. The International Journal of Advanced Manufacturing Technology, vol. 36(9-10), 2008, pp. 833-843.
- [25] C. Bock, X. Zha, H. W. Suh, & J. H. Lee, Ontological product modeling for collaborative design. Advanced Engineering Informatics, vol. 24(4), 2010, pp. 510-524.
- [26] S. Mostefai, A. Bouras & M. Batouche, Effective collaboration in product development via a common sharable ontology. International journal of computational intelligence, vol. 2(4), 2005, pp. 206-212.
- [27] J. Lee, H. Chae, C. H. Kim & K. Kim, Design of product ontology architecture for collaborative enterprises. Expert Systems with Applications, vol. 36(2), 2009, pp. 2300-2309.
- [28] K. Y. Kim, D. G. Manley & H. Yang, Ontology-based assembly design and information sharing for collaborative product development. Computer-Aided Design, vol. 38(12), 2006, pp. 1233-1250.
- [29] X. Chang, L. Zheng & J. Terpenney, Ontology development and optimization for data integration and decision-making in product design and obsolescence management. In Ontology Modeling in Physical Asset Integrity Management, 2015, pp. 87-132. Springer.
- [30] H. Molet, E. Ballot, J. Dutreuil & F. Fontane, Systèmes de production et de logistique. Hermès Science. 2006.
- [31] N. F. Noy & D. L. McGuinness, Ontology development 101: A guide to creating your first ontology. 2001.
- [32] M. Horridge, S. Jupp, G. Moulton, A. Rector, R. Stevens & C. Wroe, A practical guide to building owl ontologies using protégé 4 and co-ode tools edition1. vol. 2. The university of Manchester, 2009, pp. 107.
- [33] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, S. B. Groszof & M. Dean, SWRL: A semantic web rule language combining OWL and RuleML. W3C Member submission, vol. 21(79), 2004, pp. 1-31.
- [34] I. Horrocks, P. F. Patel-Schneider, S. Bechhofer & D. Tsarkov, OWL rules: A proposal and prototype implementation. Journal of web semantics, vol. 3(1), 2005, pp. 23-40.
- [35] M.J. O'Connor, & A. K. Das, SQWRL: a query language for OWL. In OWLED, vol. 529, No. 2009.