Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 657960, 10 pages
http://dx.doi.org/10.1155/2013/657960

Research Article

Port-Based Ontology Modeling for Robot Leg Conceptual Design

Dongxing Cao,[1] Ming Wang Fu,[2] Jiajia Mi,[1] and Zhanwei Li[1]

[1] *School of Mechanical Engineering, Hebei University of Technology, Tianjin 300130, China*
[2] *School of Mechanical Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong*

Correspondence should be addressed to Dongxing Cao; dongxingcao@gmail.com

Received 12 April 2013; Accepted 12 September 2013

Academic Editor: Shengfeng Qin

Port, as an interface of the component, can be used to exchange information with its outer environment. It is the basis of the component concept and configuration generation. In this paper, port-based ontology representation is first given, and component knowledge and the primitive concepts are presented to describe robot leg ontological concepts. Secondly, the taxonomy of port-based ontology is built to map the component connections and interactions to cluster functional blocks, and their semantic synthesis is employed to describe component ontology. Next, an approach is proposed for computing semantic similarity by mapping terms to function ontology and by examining their relationships based on port ontology language. Furthermore, the construction process of port-based ontology concepts is described and its elements are related to the similarity between two concepts. Finally, a robot leg example is shown to illustrate the proposed approach.

1. Introduction

A port is viewed as the location of intended interaction of two components or a component and its environment [1]. It plays an important role in the component concept generation. It constitutes the interface of a component and defines its boundary. Singh and Bettig [2] defined assembly ports as one or more low-level geometric entities that undergo mating constraints to join parts. They adopted the port-based composition to describe the hierarchical configurations of complex engineering design and realized assembly design through determining port compatibility and connectability. Breedveld [3] described a port as the "point" of interaction of a system, subsystem, or element with its environment in order to realize the port-based modeling of dynamic systems on the basis of bond graphs. Campbell et al. [4] developed a functional representation based on the ports of connectivity with other components to describe how energy and signals are transformed between ports. Horváth et al. [5] defined a port as the place of action for a physical effect. Based on energy flow, they classified contact ports into inports and out-ports and considered certain physical effects as occurring inside the objects and others as occurring between the objects. To formalize port descriptions, ontologies are introduced to be used for port representation, in which the classes include the ports themselves along with the attributes that allow designers to define the ports. These classes are a subset of artifact ontology, which can describe not only the interface, but also the internal characteristics of components and subsystems. Also, they adopted design concept ontology as a comprehensive methodology for managing conceptual design, including structure, shape, and functionality. Ozawa et al. [6] proposed a common ontology to support different information-level sharing between humans and multiple modeling and simulation software agents. Unified taxonomies and keyword networks can be built to support model retrieval and repository management available to designers in these domain ontologies [7]. In addition, there has been significant research into functional representation. Stone and Wood [8] presented the concept of functional basis—a formal function representation and a standardized set of function-related terminologies to support functional modeling—which consists of function and flow sets. Any functions can be described in the form of simple function sets. Furthermore, when the functional structure of a product is built, different functional classifications can be identified based on the functions themselves. Functions may be used for conveying the designers' intent. This is illustrated

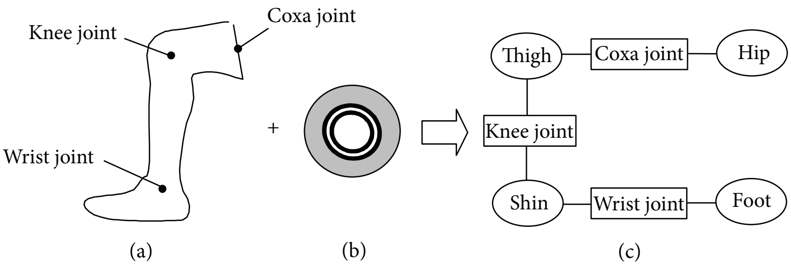


Figure 1: Leg and wheel structure corresponding graph representation.

in the design process developed by Kirschman and Fadel [9]. They presented the taxonomy of elemental mechanical functions and derived four basic types of functions, which are related to the concepts of motion, power/matter, control, and enclosure, of which each can be used with many decomposition techniques. De Kleer [10] defined function as a causal pattern between variables. The functional symbols in the natural language with the "verb + noun" style represent the designers' intention. Ontology representations not only convey and encapsulate both syntax and semantics, but also allow computer programs to share, exchange, extend, reuse, and translate information. The representations can be based on either frame-based logic or description logic.

An approach to port-based ontology that primarily focuses on performing the activity for design concept is proposed in this paper. It is difficult to build an appropriate configuration for a specific product if the developed product is not yet known. Thus, there is a tremendous need to develop an effective technology that can capture component concepts that are involved in product development. The proposed port-based ontology attempts to address this issue. The remainder of this paper is organized as follows. Section 2 gives a port-based ontology representation. Section 3 establishes component ontology knowledge base. Section 4 presents function ontology concepts, gives port attributes and taxonomies in a hierarchy, and discusses port ontology concept measures. Port-based ontology modeling process is described in Section 5. A case study and analysis of results are presented in Section 6. Finally, concluding remarks and further research are given in Section 7.

2. Port-Based Ontological Representation

A port can be viewed as the joint of human body. And it properly connects each part and makes it collaboratively work. It plays an important role in functional design of robot leg system based on port description. The total function characterizes general purpose of the designed product [11]. It may need to be decomposed into a set of subfunctions in the hierarchy. During this phase, we should carefully define component interfaces with ports and specify the associated port form.

Figure 1 gives a graph representation with leg and wheel configuration, in which Figure 1(a) presents three main joints of the leg: coxa joint, knee joint, and wrist joint. They have different functions corresponding the ports, respectively. Each port plays a different role in implementing leg's motion. Figure 1(b) indicates a wheel, and Figure 1(c) gives graph representation for robot leg system. However, it is actually difficult to go walking as human being from a technique standpoint. Thus, it is feasible to combine leg with wheels together for the robot leg. As a matter of convenience, we focus on establishing design components for implementing the specific function according to port-based ontological description.

Assume that a system (S) consists of n ports and it has m connectors (CON). Therefore, we can formally represent a product system as follows:

$$\begin{aligned} S &= \sum_{k=1}^{n} \sum_{l=1}^{m} P_k \mathrm{CON}_l \\ \mathrm{CON}_l &= \mathrm{INT}\left(\mathrm{CO}_i, \mathrm{CO}_j\right) \quad i \neq j, \end{aligned} \tag{1}$$

where P means ports, which exist in between two components or simple component; CO indicates the components, and INT stands for the interaction between components, in which i, j, k, l, n, and m are the positive integers.

In (1), CONs represent the action between two components. They can realize some specific functions, such as "fastening parts," "transmitting torques," "bounding parts". Sometimes they constitute the prototype components.

3. Knowledge Base Establishment

3.1. Port Compatibility Rules. A component is a design object with a complete specification describing how it is connected to other components in a configuration. For example, a shaft is assembled into the wheel by an axle link or two shafts are linked by a shaft coupling. They collectively form a port or an interface with each other. Then, a configuration is generated when two or more components are connected with each other via their interface as shown in Figure 2.

Compatibility checking occurs when two component models are connected. The port-based description can be conducted by the logic representation, which verifies whether all the attributes of the two ports satisfy the compatibility requirements specified in the port ontology. In a logic description, the port definitions and compatibility rules are stored in the knowledge base which is a collection of axioms for describing the true conditions of the port connection domain. When a port is established, the system queries the

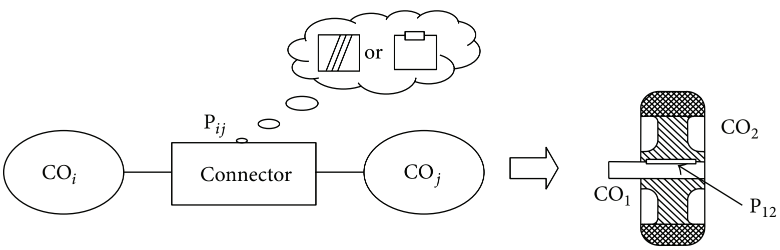
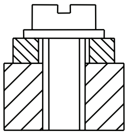
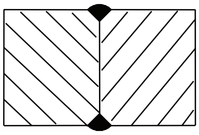
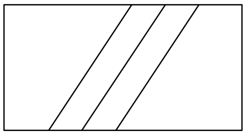
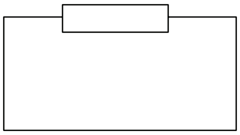
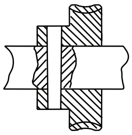


Figure 2: Port model between two components.

Table 1: The static connector types of two components.

Names	Fastener link (FK)	Weld link (WL)	Shaft coupling (SC)	Key link (KL)	Pin link (PL)
Types					

logic representation to verify if the connected port instances satisfy all the rules. For example, to connect a circular-hole port, the port must have either a pin shape or an axle shape form attribute. A system can be established as a configuration of components by connecting the components at their ports. At the same time, the connected two ports need to be compatible, or else they do not connect together; for example, a square plug does not fit a round hole. In this paper, we illustrate how the port ontology can be used to define general rules for port compatibility. The port ontology, has been defined, but it does not include the concept of compatibility yet. We can use the phrase named "is-compatible-with" to identify the port types that are compatible with each other [12]. For example, we can define the compatibility of a circular-hole port whose rule explicitly specifies that only the axle port and pin port can be connected to a circular-hole-port. This research defines the port topological properties (PTP) as a flexible mechanism to link and configure different ontologies into larger ones.

Assume that X represents the set of components in a product and a relation R_{port} can be defined in such way that it denotes port compatibility below

$$P = xR_{\text{port}}y \quad \left(x, y \in X;\ R_{\text{port}} \in \text{rules}\right), \qquad (2)$$

where x and y are components in X. R_{port} stands for a compatibility; that is, they exist in relativity, which contains an equivalent relation, a public relation, an inclusion relation, and a transfer relation. These relations are defined as follows.

3.1.1. Equivalent Relation. If x and y have the same port type and port attribute, namely, $x \equiv y$ in mathematics, they are of compatibility and can form a mutual port, that is, $xR_{\text{port}}y$. For instance, key link, pin link, shaft coupling, and so forth are often viewed as a kind of typtical equivalent relations shown in Table 1.

3.1.2. Public Relation. If x and y have the public port type and port attribute, then $x \cap y \neq \emptyset$ can be defined from a mathematical perspective. They are also compatible and can form a shared port, that is, $xR_{\text{port}}y$. For instance, common components or parts in a product system are generally viewed as public relation.

3.1.3. Inclusion Relation. If the port types and port attributes of x completely belong to y and are unreversed, then it can be represented as $x \subset y$ and $y \not\subset x$. They are also compatible and form an oriented port, that is, $xR_{\text{port}}y$. For instance, some components with support functions or packing functions can be considered as inclusion relation.

3.1.4. Transfer Relation. If x, y, and z satisfy $x \subset y$ and $y \subset z$, then $x \subset z$; the ports x, y, and z will be of conduction attribute, namely, $xR_{\text{port}}yR_{\text{Port}}z$. For instance, worm gear, cone gear, rack pinion, and so forth are often viewed as transfer relation shown in Table 2.

The connectors are the basis of connecting two components to implement the specific relations. There are number of connectors which have different functions, such as motion, orientation, and dynamic requirements. There are two kinds of connectors, that is, static connectors and kinetic connectors. Table 1 presents the static connectors, such as fastener link, weld, shaft coupling, key link, and pin link. Meanwhile, Table 3 gives the kinetic connectors, such as worm gear, cone gear, gear mechanism, belt gear, and rack pinion. They constitute a set of connectors for the knowledge base of alternative design.

The above-mentioned relational rules are solely based on port names and their attributes. However, if a new port class is added to the port ontology, it is not suitable for only using port names. This needs to update compatibility rules. An effective measure is to use attributes to present the compatibility constraints. A circular port can be connected between shaft and gear with similar geometric features as shown in Figure 2. When foot wheel contacts the earth surface, a port is formed. If robot leg walks along on the ground, then a drive is forced on a shaft which generates a port between shaft and

Table 2: The kinetic connector types of two components.

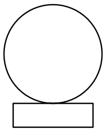
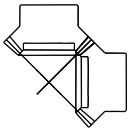
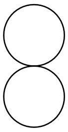
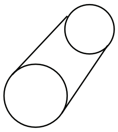
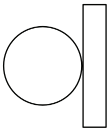
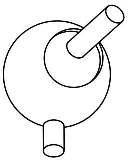
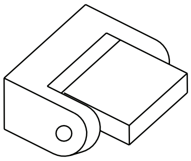
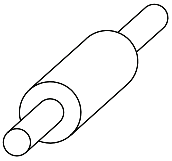
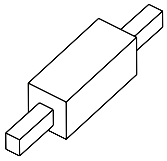
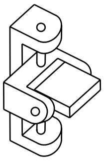
Names	Worm gear (WG)	Cone gear (CG)	Gear mechanism (GM)	Belt gear (BG)	Rack pinion (RP)
Types					

Table 3: The prototype components of constraint joints.

Names	Spherical joint (SJ)	Revolute joint (RJ)	Cylindrical joint (CJ)	Prismatic joint (PJ)	Cross-pin joint (CP)
Types					
Dofs	3	1	2	1	2

wheel. We could express this rule using low-level geometric constraints on the type and dimensions of port features. If two components are compatible, they certainly possess relativity. We can also evaluate the compatibility of both components by evaluating the semantic relativity [13].

3.2. Prototype Component Base. The prototype component corresponds to the basic function unit, and it cannot be separated further. Port is used to configure a particular component and is restricted within the configuration interface for the component. Multijoints are the basis unit for constructing robot leg system. General joints are described as follows [14].

A spherical joint, shown in column one in Table 3, is satisfied with the condition that the center of the ball coincides with the center of the socket. Then, terminal equations are three scalar constraint relationships that restrict the relative positions between the components in the following:

$$\left[e^x \ \ e^y \ \ e^z\right]^T \delta_{r3} = \left[0 \ \ 0 \ \ 0\right]^T, \tag{3}$$

where e^x, e^y, and e^z are unit vectors and δ_{r3} represents the relative virtual displacement between the components. Essentially, three constraint relationships are imposed, and there is no relative displacement at the joint in the x-y-z directions, but three virtual angular rotations are free. Therefore, the dof equates to three.

A revolute joint, shown in column two in Table 3, is constructed with bearings which only allows relative rotation about a common axis in a pair of components but precludes relative translation along the axis. The five terminal constraint equations are

$$\left[e^x \ \ e^y \ \ e^z\right]^T \delta_{r3} = \left[0 \ \ 0 \ \ 0\right]^T \qquad \left[e^x \ \ e^y\right]^T \delta_{\theta 2} = \left[0 \ \ 0\right]^T, \tag{4}$$

where $\delta_{\theta 2}$ represents the relative virtual angular rotation between the components and dof equates to one.

A cylindrical joint between components is shown in column three in Table 3. It permits relative translation and relative rotation between two components along a common axis. The four terminal constraint equations are

$$\left[e^x \ \ e^y\right]^T \delta_{r2} = \left[0 \ \ 0\right]^T \qquad \left[e^x \ \ e^y\right]^T \delta_{\theta 2} = \left[0 \ \ 0\right]^T. \tag{5}$$

A prismatic joint, shown in column four in Table 3, allows relative translation along a common axis between two components but precludes relative rotation about the axis. The five terminal constrained equations are

$$\left[e^x \ \ e^y\right]^T \delta_{r2} = \left[0 \ \ 0\right]^T \qquad \left[e^x \ \ e^y \ \ e^z\right]^T \delta_{\theta 3} = \left[0 \ \ 0 \ \ 0\right]^T. \tag{6}$$

Above-mentioned constraint joints are shown in Table 3. Also, the other connectors can be obtained, and they are the common connecting mechanisms shown in Table 3. These components can be put into knowledge base for indexing and reusing.

4. Functional Ontology Concepts

4.1. Port Attributes and Port Taxonomy. In mechanical system, common components, such as gear, bearing, fastener, gimbal, shaft coupling, and spring, are named the prototype components. They are the basis of prototype concept generation. And they can be well defined by a set of structural attributes. The connector is defined by an action of two basic mechanisms with physical law or geometric axiom, and it provides constraint on them to implement the specific functions. On the basis of constraint characteristic, mechanical products can be classified into three kinds of form, that is, static constraint, kinetic constraint, and motional force transfer as shown in Table 4. The physical effect of the connectors has Hooke's law, Newton's law, friction principle, and so on. Their representation includes geometric constraint or primitive components, named mechanical standard parts, such as fastener, spring, and bearing. They are referred to as the standard connectors in mechanical engineering field.

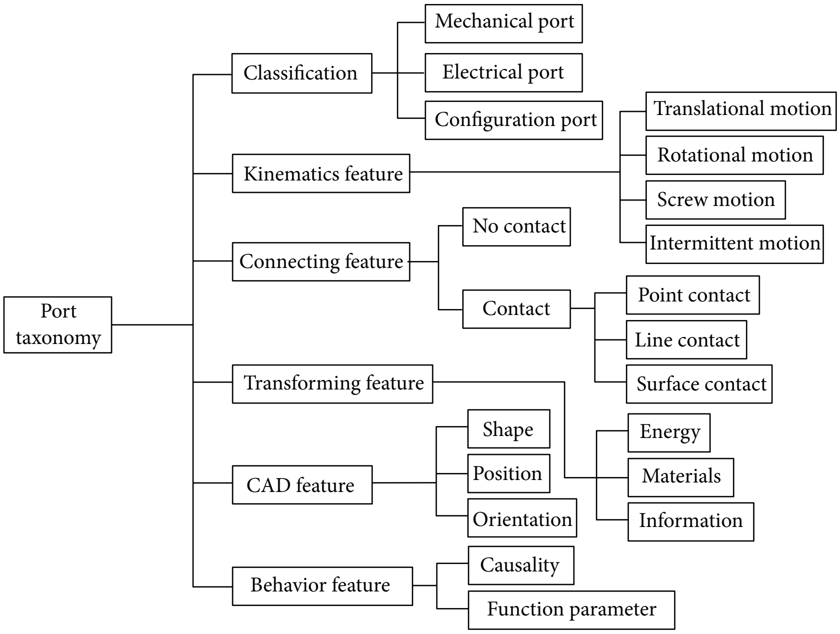


Figure 3: Port taxonomy.

Table 4: Connectors of mechanical components.

Functions	Behaviors	Types of connectors
Fastener connectors	Fastening	Fasteners
	Jointing	Binder
Kinematic constraint connectors	Rolling	Bearing
	Sliding	Slide rail
Force transfer connectors	Airproof	Sealing ring
	Speed change	Gear drive

In general, port attributes include function, behavior, and structure, in which function attribute contains port classification, connecting form, and kinetic and transforming way. Behavioral attribute indicates causality and physical parameters. And structural attribute means shape, position, and orientation. They are the basis of port ontology modeling, and their taxonomies are shown in Figure 3.

4.2. Port Ontology Concept Measures. It is easy to obtain an ontology concept by using attribute representation of port ontology. Therefore, it is very important to distinctly analyze port attributes before naming concept ontology. Attribute representation of a port is shown in Figure 3. This taxonomy allows the users to quickly find components in an ontology library by mapping operation, which contains component models and an alternative way to access components in the library. For example, two mechanical contacted parts have the same attribute of transferring mechanical energy, and they easily form a mechanical port.

Prototype concepts are the basic unit of function concepts and are interdependently defined with each other. They are defined by using a set of prototype terms and viewed as a semantic description of functional elements. A connector is defined as the interaction between two components and it is the location of generating physical effect [1, 15].

As the phrase of "verb + noun" is used to represent the function, the semantic description of functional elements can be quantitatively obtained by indexing its verb and noun terms. Therefore, functional concept (δ_i) is related to verb (V_i) and noun (N_i) and represented as follows:

$$\delta_i = \{V_i, N_i \mid V_i \subset \text{Verbs}, N_i \subset \text{Nouns}\}. \tag{7}$$

Specifically, the similarity (SIM) between two prototype concepts can be represented as follows:

$$\text{SIM}\left(\delta_i, \delta_j\right) = \frac{\text{The number of matching terms with each}}{\text{The total number of terms}}, \tag{8}$$

where δ stands for the prototype concept; that is, (δ_i, δ_j) refers to two different prototype concepts.

The connectivity of different prototype concepts can be quantitatively measured on the basis of the similarity calculation [13]. Based on the function, four typical connections are identified among primitive design concepts in the following [5].

(i) *Cause Connection* (CC). A design concept necessitates the function delivered by another design concept to achieve a needed function. It has a feature with transfer relation. For example, a gear can realize rotation from the other gear drive.

(ii) *Equal Connection* (EC). If two design concepts present the same function based on similar or dissimilar constituents, such as entities, situations, and phenomena, then they form equal connectors. It has a feature with equal relation. For example, fastener link and weld are two kinds of equal connectors.

(iii) *Share Connection* (SC). Simultaneously possesses one ontology concept when SC distributes in different locations and spaces with several functions. It has a feature with public relation. For example, a concept of fastener ontology is used to share static connector for different components.

(iv) *Bind Connection* (BC). BC expresses the assertion that there is no interdependence between two or more design concepts. It has a feature with inclusion relation. And the bind connection of the design concepts will depend on the constraints' relationships.

Four form connections are the foundation of generating module concepts. A module can be formed by combining different connections with more components. In the process of primitive concept acquisition, it is possible to distinguish properties from the specific domain. These properties can be explored by describing port ontology. In the practical application, these distinctions refer to groups of properties that are known as port concept. For example, a robot leg system can be viewed as several functional modules, and each module exists in many components and connectors. Identifying and separating these basic connections will be important for structuring a new prototype concept in port-based ontology. It can give rise to a strong internal connection or a weak coupling connection. According to the above analysis, four kinds of connection relations can be uniformly represented as follows:

$$\mathrm{CON} = \left\{\mathrm{INT}\left(\mathrm{CO}_i, \mathrm{CO}_j\right) \mid i \neq j; \mathrm{INT} \subset \mathrm{CC}, \mathrm{EC}, \mathrm{SC}, \mathrm{BC}\right\}. \tag{9}$$

To determine the connection degree between primitive concepts, the similarity degree (SIM) is defined as the similarity evaluation of two primitive concepts given in (8). In the practical application, the ontological relation between concepts can be determined by using the primitive concept classification and a set of familiar attributes. For example, the robot leg is the basic mechanism to implement robot walk, and it can go ahead, back, and get across the obstacles. It plays an important role in determining robot joints and components to construct port-based ontology, that is, to obtain a stronger inner connecting or a weak coupling relation.

5. Port-Based Ontology Modeling

The functions are decomposed by matching the primitive function concepts of knowledge base. They describe the physical effects corresponding the port connectors. Port-based ontological semantics also describe connection information and structure information between primitive components. Primitive functions or subfunctions are aggregated into the total function in a bottom-up manner. Behavioral description specifies the connectivity and causality which constitutes a hierarchical semantic net by behavioral semantic mapping and it makes efforts to bridge the gap between port function and port structure [16]. Configuration describes which components are involved and whether they are mapped or interact with each other. They specify the relations of component position, topology, and kinematics. The positional relation quantitatively describes how the artifact is positioned and oriented in a three-dimensional space presented in (3), (4), (5), and (6). The topological relation interprets the condition of their physical connections. Contacts can be direct or indirect. Contacts can also be further specified among the individual points on the surface, domains on the surface, and complete the surface [5]. Configuration is identified by primitive form mapping in a hierarchy.

The attributes are lower-level concepts for defining ports. The attributes are divided into three main categories: function, behavior, and structure [17]. When a port is defined by function attributes, its attributes describe the intended use of the port. Port attributes are crucial to constructing the concept ontology process. Therefore, it is very important to distinctly analyze port attributes before defining concept ontology.

As the language is governed by grammar or a set of rules, it is possible to algorithmically process language to identify patterns and extract information [18]. In English language, verbs and nouns or more generally noun phrases are used. Their grammar functions can be either the subject or the object of the verb, and the typical construction of English sentences is subject-verb-object (SVO). We use this language model to identify verbs to connect engineering lexicons.

We defined port concepts with intention-rich functional concepts. However, most of "verb + noun" phrases often slack such intention of functional representation. They have no machine understandable definition of concepts. In the function behavioral structure (FBS) model, the function symbol in natural language in the verb + noun style represents the intention of designers. We tried to identify this kind of function concepts as hiding in function semantic structure. A function of a component cannot be determined until the component is installed in a specific system with a specific configuration [2]. Although a function of a component depends on other components, the description itself should be local. In such cases, a group of functions is sometimes formed by a set of components across the intersubsystem boundary.

Function semantic expression from users' requirements describes the process of port ontology generation. In the following illustration, the semantic expressions of a robot leg system are extracted from the users' design description and are given as follows.

> Run up thigh to add the height of body, while putting down thigh to reduce the height of body.
>
> Leg is a part of robot body, and also thigh is a part of leg.
>
> Foot wheel is the foot of robot, and it can make the robot walk.
>
> Drive KJ-motor to reduce the span in a clockwise direction.

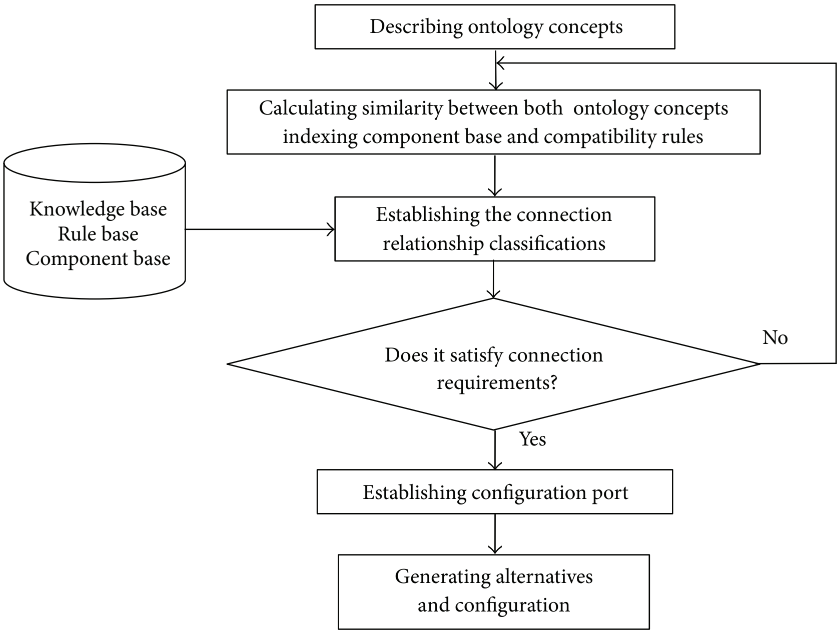


FIGURE 4: Port-based ontology modeling process.

Drive WJ-motor to make foot wheel rotation.

Knee joint has the function of driving shin rotation.

Connect WG1 with hip together.

Assemble key link to shaft and foot wheel.

Fix WJ-motor to the end of shin.

Assemble shaft into bevel gear.

...

Return.

These semantic expressions include a phrase of "verb + noun", that is, "is_a" link, "part_whole" relation, "has_function" relation, and so on. "Noun" and "noun phrase" are composed of the keywords of function concepts [19]. Actually, there are different types of artifact ontologies. They are firstly filtrated by several rules. Secondly, the selected artifacts are included and subsequently are specialized to an engineering domain by instantiation, term, and concept mappings and additional specific axioms. Finally, a larger ontology is synthesized by the compatibility to connect corresponding artifact ontologies. We developed formal steps to implement port-based ontology modeling shown in Figure 4 as follows:

(i) describe the component functions to build port ontology and lexicon;

(ii) pick up the various semantic relations to formalize users' requirements for regular term arrangement;

(iii) establish a hierarchical functional concept to conveniently obtain primitive concepts;

(iv) measure the shortest path between two functional concepts by using the semantic similarity;

(v) distinguish compatibility port ontology to cluster into a fit component configuration.

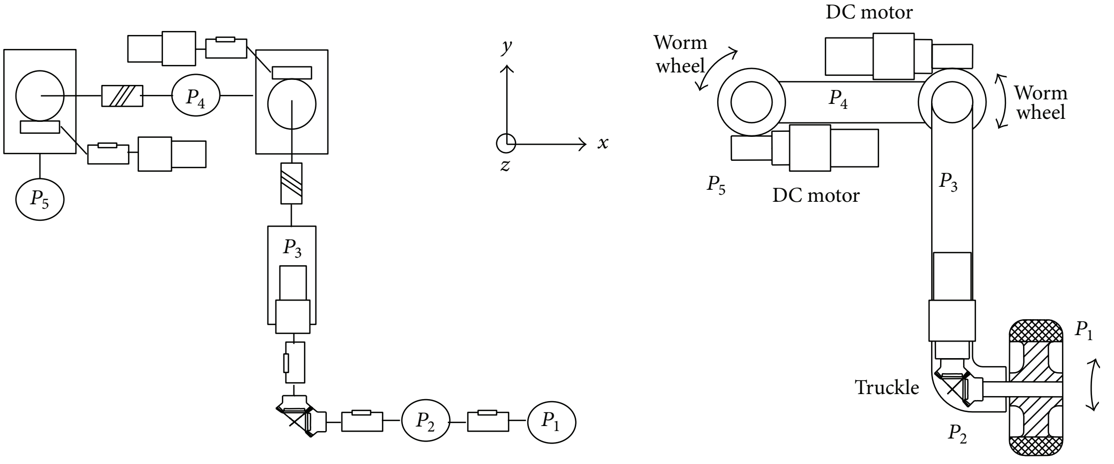
6. Case Study and Result Analysis

The leg is a part of robot body and it is the main component to realize robot motion. The dofs determine the complexity of robot motion and routing problem. It is a much familiar and simple robot leg for two dofs and foot bottom truckle. It is easy to realize for simplifying leg motion paths. According to users' requirements and human being's leg structure, three joints, called ports, are needed to realize leg's functions as shown in Figure 1. Thigh and hip are linked by coax joint, and thigh and shin are linked by knee joint. Five basic components $P_1 \sim P_5$ are easily established to configure robot leg. Figure 3 presents the connector model of two components for robot leg. The connector C_{12}, that is, key link, connects between P_1 and P_2, since P_1 and P_2 are common linearity with each other. We select one of the static connectors, such as SC, KL, and PK, as shown in Table 1. In addition, as P_1 and P_2 locate on different surfaces, it is a fit choice for KL. It can be obtained from knowledge reasoning through indexing component base. So, a key link will be selected to satisfy the robot leg requirements. Also, port P_{23} will be generated between shin and shaft. As P_2 and P_3 consist in the relationship of plane motion, we select kinetic and static connectors, that is, CG and KL, as shown in Tables 1 and 2. All ports will be obtained by semantic description and knowledge reasoning on account of port-based ontology as shown in Table 5. The configuration relationships of ports and connectors are shown in Figure 5(a).

The rotation is the main function that joint legs realize. It is also a main way for implementing the robot leg functions. An additional function is needed to realize the robot leg

Table 5: The verbs and nouns of describing robot leg port concepts.

Ports	Port generations	Verbs	Nouns	Examples
P_{12}	(shaft, foot wheel)	Assemble, rotate, connect, transform	Shaft, foot wheel, key link	Assemble shaft to foot wheel
P_{23}	(Shin, shaft)	Transform, connect, fasten	Shin, shaft, WJ-motor, CG, key link	Transform torque from WJ-motor to shaft
P_{34}	(thigh, shin)	Connect, rotate, assemble	Thigh, shin, WG2, shaft coupling, key link, KJ-motor	Fasten KJ-motor to worm through key link
P_{45}	(hip, thigh)	Fasten, assemble, has, is_a	Hip, thigh, WG1, shaft coupling, key link, CJ-motor	Fasten WG1 to hip through bolted connection



(a) (b)

Figure 5: A robot leg configuration.

functions, such as the motion state of the thigh, the motion state of the shin, or the additional mechanism to realize the robot motion. In doing so, a set of phrases and relations are established to describe additional functions. For example, rotate thigh, move hip, rotate foot wheel, drive CJ-motor, and so forth. These phrases can describe the function of a robot and realize the requirements of robot motion. Thus, the function concepts of robot are obtained. And they have the corresponding port functional semantic attributes, as shown in Table 5.

To effectively realize the function of the robot leg, according to design requirements, we will transversally extend total function of the robot leg into several function modules. They form a port-based ontology structure in which each function is described by using different ontology concepts, such as "verb +noun" drawn from design requirements [20].

Different function concepts are represented on the basis of "verb + noun" phrases. Some relationships should be extracted, such as is_a, part_whole, has_function, and has_feature [13]. At present, we have only collected the function taxonomy of robot leg. Four types of relationships are described in Table 6. The number of concepts depends on the users' requirements and the definition of concepts. The standard worksheets have been developed to easily acquire port ontology and lexicon. At the same time, they can automatically upload the required data into the Protégé editor [18], which is one of the most widely used ontology editors. Protégé provides a visual tool for port ontology editing, including concept, taxonomy, and relationship building along with ontology visualization.

Figure 5 presents the process of clustering concepts into a configuration of the robot leg. It calculates the correlations between the matched concepts of terms to determine the connection of two concepts. For a set of concepts, a concept is highly correlated with others if it is less distant from them, that is, semantically closer, or contains more words that match the specific terms, that is, are lexically closer [7]. At the same time, component generation depends on the function taxonomy. Figure 5(a) gives different weight scores and their concept similarity measures results corresponding to robot leg configuration which contains 6 function concepts, 18 function terms, and 9 components for clustering robot leg. Figure 5(b) presents the structure of robot assembly. Table 7 gives different function concepts corresponding to the primitive connectors, attributes, and port types as part of the functional semantic description after the designers

Table 6: Classification of the relationships.

Relationships	Concepts	Definitions	Examples
Is_a	Key link, connector	Relationships between parent and son or special and general	Is_a (key link, connector)
Part_whole	Thigh, robot leg	Relationships between part and whole	Part_whole (thigh, robot leg)
Has_function	CG, foot wheel	Refer to the connection between two ontology concepts	Has_function (CG, foot wheel)
Has_feature	Worm wheel, worm	Physical attributes or geometric attributes	Has_feature (worm, worm wheel)

Table 7: Measure between two function concepts.

Function concepts	Primitive connectors	Causality	Attributes	Port types
Transform motion/realize rotation	Key link	EC, SC	Surface contact	Mechanical port
Drive wrist joint motion/realize rotation	WJ-motor, CG, key link	CC, EC	Line contact, surface contact	Mechanical port
Drive knee joint motion/move shin	WG2, shaft coupling, key link, KJ-motor	CC, BC, EC	Line contact, surface contact	Mechanical port
Move thigh/drive coxa joint motion	WG1, shaft coupling, key link, CJ-motor	CC, BC, EC	Line contact, surface contact	Mechanical port

have made use of effective semantic inference. The causality indicates that they include cause connection, equal connection, share connection, and bind connection for two function concepts.

7. Concluding Remarks

This paper describes port-based ontology modeling for robot leg concepts, determines port types, and extends port attributes in a hierarchy. One of the main goals of the research was to clarify the relationship of component ontology concepts associated with functionality, that is, is_a relation, part_whole relation, and has_function relation. Although the functional decomposition trees can be used to represent the scheme design, the approach often leads to combinational explosion. In this paper, the semantic similarity approach is applied to port-based ontology and specified by users to enable the system to generate various function modules. Port-based ontology may be used in the conceptual design of the electromechanical system by providing the function module; that is, it can quantitatively realize semantic measures and effectively build function modules. Function concepts and corresponding terms are related to design requirement descriptions. However, further research needs to consider the following.

(i) To get the correct results of semantic measures, the accuracy of function concept description and port-based ontology information retrieval are very important. In most cases, the number of components, function concepts, and terms is not completely corresponding to each other, and they exist in one-to-many or many-to-one relationships. Axiomatic design should be adopted to clarify these relationships.

(ii) The paper only focuses on a simple robot leg conceptual design; however, actually it is more complicated about conceptual design of a complex product, and it needs to build some fit modules through using effective clustering algorithm. It is worth further investigating in the future.

(iii) Each designer has different functional semantic description for the same product; perhaps they obtain a different function ontology concept and further generate the other alternatives. This depends on designers' backgrounds, educations, and preferences. Therefore, we will focus on them in the future.

Acknowledgments

This research is sponsored by the National Nature Science Foundation of China (NSFC) under Grant no. 50775065 and no. 51275152 and Nature Science Foundation of Hebei Province under Grants E2008000102 and E2013202123.

References

[1] V. Liang and C. J. J. Paredis, "A port ontology for conceptual design of systems," *Journal of Computing and Information Science in Engineering*, vol. 4, no. 3, pp. 206–217, 2004.

[2] P. Singh and B. Bettig, "Port-compatibility and connectability based assembly design," *Journal of Computing and Information Science in Engineering*, vol. 4, no. 3, pp. 197–205, 2004.

[3] P. C. Breedveld, "Port-based modeling of dynamic systems in terms of bond graphs," in *Proceeding of the 5th Vienna on Mathematical Modeling*, Vienna, Austria, February 2006.

[4] M. I. Campbell, J. Cagan, and K. Kotovsky, "Agent-based synthesis of electromechanical design configurations," *Journal of Mechanical Design*, vol. 122, no. 1, pp. 61–69, 2000.

[5] I. Horváth, K. Dorozsmai, and V. Thernesz, "A feature-object-based practical methodology for integration of conceptual and morphological design," in *Proceedings of the Lancaster International Workshop on Engineering Design (CACD '94)*, pp. 131–149, Lancaster University EDC, Lancaster University, 1994.

[6] M. Ozawa, M. R. Cutkosky, and B. J. Howley, "Model sharing among agents in a concurrent product development team," in *Proceedings of the Workshop on Knowledge Intensive CAD, KIC-3, IFIP5. 2 Working Group*, Tokyo, Japan, 1998.

[7] Z. Li, V. Raskin, and K. Ramani, "Developing engineering ontology for information retrieval," *Journal of Computing and Information Science in Engineering*, vol. 8, no. 1, pp. 1–13, 2008.

[8] R. B. Stone and K. L. Wood, "Development of a functional basis for design," *Journal of Mechanical Design*, vol. 122, no. 4, pp. 359–370, 2000.

[9] C. F. Kirschman and G. M. Fadel, "Classifying functions for mechanical design," *Journal of Mechanical Design*, vol. 120, no. 3, pp. 475–490, 1998.

[10] J. De Kleer, "How circuits work," *Artificial Intelligence*, vol. 24, no. 1–3, pp. 205–280, 1984.

[11] Y. Kitamura, T. Sano, K. Namba, and R. Mizoguchi, "A functional concept ontology and its application to automatic identification of functional structures," *Advanced Engineering Informatics*, vol. 16, no. 2, pp. 145–163, 2002.

[12] C. J. J. Paredis, A. Diaz-Calderon, R. Sinha, and P. K. Khosla, "Composable models for simulation-based design," *Engineering with Computers*, vol. 17, no. 2, pp. 112–128, 2001.

[13] D. Cao, K. Ramani, M. Wang Fu, and R. Zhang, "Port-based ontology semantic similarities for module concept creation," in *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE '09)*, pp. 141–150, San Diego, Calif, USA, September 2009.

[14] D. Cao, W. Feng, P. Qu, and C. Yu, "Port-based ontology modeling for robot leg configuration design," in *Proceedings of the 10th International Conference on Frontiers of Design and Manufacturing*, Chongqing, China, June 2012.

[15] D. Cao, Y. Gu, M. W. Fu, and H. Jia, "Port-based ontology modeling for product conceptual design," in *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE '08)*, pp. 163–172, New York, NY, USA, August 2008.

[16] J. B. Dahmus, J. P. Gonzalez-Zugasti, and K. N. Otto, "Modular product architecture," *Design Studies*, vol. 22, no. 5, pp. 409–424, 2001.

[17] U. Roy, N. Pramanik, R. Sudarsan, R. D. Sriram, and K. W. Lyons, "Function-to-form mapping: model, representation and applications in design synthesis," *CAD Computer Aided Design*, vol. 33, no. 10, pp. 699–719, 2001.

[18] S. C. J. Lim, Y. Liu, and W. B. Lee, "Product analysis and variants derivation based on a semantically annotated product family ontology," in *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (DETC '09)*, pp. 817–826, San Diego, Calif, USA, September 2009.

[19] E. Bozsak, M. Ehrig, S. Handschuh et al., "KAON-Towards a large scale semantic web," in *Proceedings of the 3rd International Conference on E-Commerce and Web Technologies*, K. Bauknecht, A. Mintjoa, and G. Quirchmayr, Eds., pp. 304–313, Springer, Heidelberg, Germany, 2002.

[20] Z. Wu and M. Palmer, "Verb semantics and lexical selection," in *Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics (ACL '94)*, pp. 133–138, Las Cruces, New Mexico, 1994.