Proc. of the Pacific Asian Conference on Intelligent Systems 2001 (PAIS 2001), 2001. (to appear)

Ontology-based Description of Functional Design Knowledge and its Use in a Functional Way Server

Yoshinobu Kitamura, Toshinobu Kasai* and Riichiro Mizoguchi

The Institute of Scientific and Industrial Research, Osaka University
8-1, Mihogaoka, Ibaraki, Osaka, 567-0047, Japan
Tel: +81-6-6879-8416, Fax: +81-6-6879-2123, E-mail:{kita,kasai,miz}@ei.sanken.osaka-u.ac.jp

Abstract

In conceptual design of engineering devices, a designer decomposes a required function into sub-functions, so-called functional decomposition, using a kind of functional knowledge representing achievement relations among functions. However, such knowledge about functionality of engineering devices is usually left implicit because each designer possesses it. Even if such knowledge is found in documents, it is often scattered around technical domains and lacks consistency. Aiming at capturing such functional knowledge explicitly and sharing it in design teams, we discuss its systematic description based on functional ontologies which provide common concepts for its consistent and generic description. We propose a new concept named "way of achievement" as a key concept for capturing such functional knowledge. Categorization of typical representations of the knowledge and its organization as is-a hierarchies are also discussed. The generic concepts representing functionality of a device in the functional knowledge are provided by the functional concept ontology, which makes the functional knowledge consistent and applicable to other domains. We also discuss development of a design supporting system using the systematized knowledge, called a functional way server. It helps human designers redesign an existing engineering device by providing a wide range of alternative ways of achievement of the required function in a manner suitable for the viewpoint of each designer and then facilitates innovative design.

Keywords:

Knowledge modeling; Knowledge sharing; Ontology; Functionality; Design support

Introduction

Engineering design can be defined as mapping from a requirement specification at the functional level into a set of attribute values of concrete things. So, functionality plays a crucial role in the conceptual design of engineering devices [Pahl 88, Keukene 91, Chandrasekaran 93, Umeda 96]. For example, a designer often decomposes a required function into sub(micro)-functions, so-called functional decomposition [Pahl 88]. As the result, a designer obtains a micro-macro hierarchy of functions, which represents that how the required (macro-)function is achieved by sub(micro)-functions, as a conceptual skeleton of the product realizing the requirement. Because there are many methods to achieve a specific function in general, designers should select an appropriate one from many alternatives. Such activity requires knowledge of how to achieve a function, which represents achievement relations among functions. We call such knowledge about functions functional knowledge. Because many inventions are based on techniques well-known in different domains [Sushkov 95], if designers can consult a wide range of such functional knowledge in different domains, it facilitates innovative design.

However, the design know-how including such functional knowledge used in the conceptual design phase is usually left implicit because each designer possesses it, although advancement of computer technologies has enabled easy access to information related to structure and/or shape of artifacts. Even if such knowledge is found in documents, it is often scattered around technical domains and improperly categorized. For example, a categorization of connection-methods found in a textbook published by an academic society is not consistent and ill structured, because it is categorized according to different properties. Moreover, the textbook includes many categorizations based on the non-fundamental characteristics which can be derived from deeper principles.

We focus on the following three reasons for such dependency on domains and such an inappropriate categorization. The first one is that different frameworks for conceptualization are used when people try to describe knowledge in different domains. We need fundamental concepts for capturing target devices from functional viewpoint.

* Currently, Faculty of Education, Okayama University

The second reason is there are several functional concepts (i.e., verbs representing function) without clear definitions as pointed out in Value Engineering [Tejima 81]. Common vocabulary is necessary for reuse of functional knowledge in different domains. Moreover, it should be machine understandable in order to manage the knowledge. Nevertheless, only a few generic functions have been proposed in functional representation research to date [Pahl 88, Hodges 92, Lind 94]. They are at a very high level of abstraction and inflexible so that it is hard to incorporate designer's intention into the functional model. On the other hand, value engineering has developed a standard set of verbs representing function. Although it is well-organized, definitions of verbs are only for human consumption. We need a rich and comprehensive vocabulary of functional concepts with operational definitions.

The last reason of inappropriate functional knowledge is that structure (organization) of viewpoint-independent and consistent functional knowledge is not fully investigated yet. When designers select a method of achievement of a required function as mentioned before, the criteria for the selection depend on his/her own viewpoints and thus are different from each other. For example, a designer may decompose the required function "to connect two objects" into "to insert a screw into a hole" and "to tighten the screw" by selecting the "screw way". The other designer, however, may prefer "hook-fit clasp" in which the objects are connected by "to hook the crook" from the viewpoint of manufacturing (assembling) of the product, because manufacturing in the case of "hook-fit clasp" is easer than that of the screw way. In conventional knowledge description, such knowledge is used to be described for each viewpoint, and the dependency on viewpoints is implicit. We need a systematic way of description of the knowledge and treatment of such viewpoints.

The needs for consistent and sharable description of functional knowledge can be summarized as fundamental and generic concepts for capturing and describing the functional knowledge. Such specification of conceptualization is generally called an "ontology" [Gruber 93, Mizoguchi 00]. In other words, we need such ontologies that guide conceptualization of artifacts from the functional point of view.

We have developed functional ontologies including a device-centered ontology and a functional concept ontology, aiming at systematization of functional knowledge for design [Mizoguchi 00, Kitamura 00]. The main role of the ontologies is to provide well-defined concepts for description of knowledge, and then to give a basis for systematization of knowledge.

This article discusses knowledge description based on those functional ontologies. Firstly, we propose a new key concept for capturing such knowledge concerning achievement of functions, called way of function achievement. Next, categorization of typical representations of the knowledge and organization as is-a hierarchies are discussed.

Next, we discuss use of the knowledge based on the ontology. We show a knowledge-based system to help human designers decompose the required function in conceptual design phase by providing various methods of achievement of the function. The ontology of functional concepts and the systematized functional knowledge play crucial roles in these systems.

Hierarchy of Functional Knowledge and Ontologies

Figure 1 shows a hierarchy of functional knowledge built on top of fundamental ontologies. The lower layer knowledge is in, the more basic. Basically, knowledge in a certain layer is described in terms of the concepts in the lower layer. Top-level ontology defines and provides very basic concepts such as time, state, process and so on. Causal ontology specifies actions and causality against teleology. Physical world ontology specifies 3D space and entity to give axiomatic physical world with a state-based modeling reflecting a special world of design in which an entity(artifact) is created from nothing. These two ontology contributes to "Symbol grounding" of higher-level concepts, that is, functional concepts.

On top of these three, physical process ontology is introduced to specify physical(natural) processes. A very primitive definition of *process* is done in the top-level ontology to cover any sequence of events in terms of state and its change. The term "Process" is confusing, since it has three different meanings; sequence/course/stage, something found in the phrase "Burning process" and

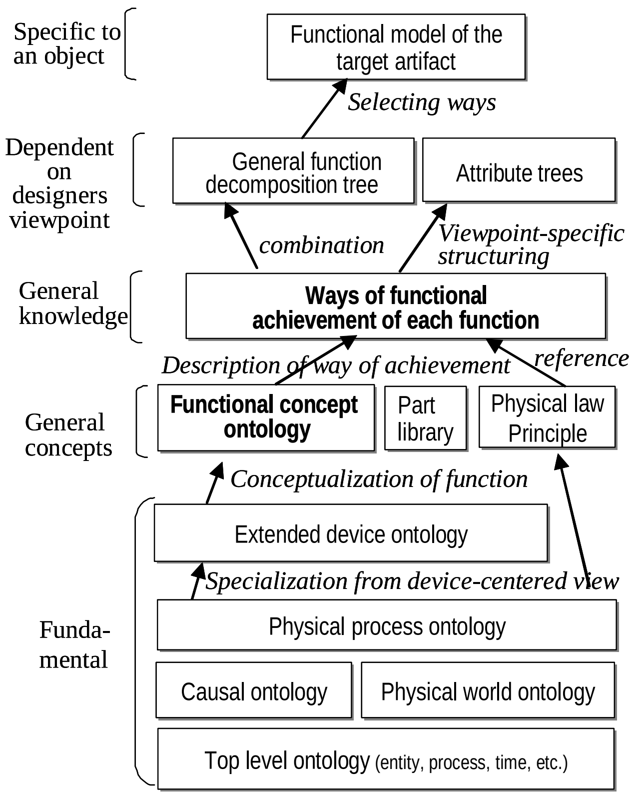

Figure 1 - Hierarchy of ontology and knowledge

processing. The first is one defined in the top-level ontology, the second is covered by the physical process ontology introduced here and the last is covered by device ontology.

The device ontology specifies the device-centered view of artifacts which regards any artifact as composition of devices which process input to produce output which is what the users need. The device ontology imposes a frame or viewpoint on physical processes in the physical process ontology to introduce an engineering perspective. That is, it introduces the concepts of a black box equipped with input and output ports. The major difference between the device ontology and the physical process ontology is that while device ontology has an *agent*, which is considered as something that plays the main actor role in obtaining the output, the physical process ontology does not have such an *agent* but has *participants*, which only participate in the phenomena being occurring. Needless to say, such an *agent* coincides with a *device* in the device ontology. Something that can be considered as that it goes through a device from the input port to the output port during which it is operated by the device is called an *object*.

Device ontology specifies the roles played by the elements that collectively constitute a device. The concept of role is a hot topic in ontological engineering because an object plays different roles in different situations, and the fact has been a major source of failure in conceptualization of the world [Mizoguchi 00].

A naïve idea of device ontology was born in system engineering. It is composed of components and connection between them and has been extensively used in many engineering areas as well as in design community as is discussed in [Pahl 88]. However, it has no criterion of which role should be played by which component and the assumptions behind the ontology is implicit, and therefore modeling of artifacts based on that ontology can be ad-hoc. Even worse, it is hard to compare it against other ontologies and its limitations are not clear. De Kleer and Brown introduced an idea of conduit into the naïve device ontology[de Kleer 84]. His ontology still leaves the identification of objects unclear. The authors propose an extended device ontology based on Ontological Engineering which includes four different concepts of behavior, introduces concept of *medium*, and covers mechanical domains also [Mizoguchi 00].

These five ontologies(Top-level, causality, physical world, physical process and device ontologies) collectively work as a substrate on which we can build consistent knowledge in layers.

Functional concept ontology [Kitamura 00] specifies functional concepts as an instance of *function* defined in device ontology. The definitions are scarcely depends on the device, the domain or the way of its implementation so that they are very general and usable in a wide range of areas. Theories and principles of physics and abstract part library also belong to this class of knowledge called *general concept layer*.

Way of functional achievement is knowledge about *how*(in what way) a function is achieved, whereas the functional concept is about *what* the function is going to achieve. The inherent structure of such knowledge is organized in an *is-a* hierarchy from which the other three structures are derived according to the requirement. The *is-a* structure is carefully designed identifying inherent property of each *way* to make it sharable and applicable across domains. One of the key issues in knowledge organization is clear and consistent differentiation of *is-a relation* from other relations such as *part-of*, *is-achieved-by*, etc. keeping what is the inherent property of the target thing in mind.

This paper mainly discusses the way of function achievement and knowledge description based on it. The implementation of the device ontology and the functional concept ontology is also discussed.

Framework of Functional Models of Artifacts

In our functional modeling framework [Sasajima 95, Mizoguchi 00, Kitamura 00], the model of artifacts consists of a behavioral model and a functional model as shown in Figure 2. According to the device-oriented ontology, "structure" of the behavioral model describes the existence of components, topological connections among them, and a micro-macro hierarchy among components and (sub-)systems. The behavioral model of components represents changes of attribute values of entities which flow through the component. The structural micro-macro relations represent hierarchical organization of behavior.

On the other hand, at the functional level, a (base-)function of a component is defined as a result of teleological interpretation of a behavior of the component under an intended goal [Sasajima 95]. Each of the base-functions plays a role for another function (called meta-function) which represents interdependency among base-functions [Kitamura 99]. The micro-macro relations among functions represent that the macro-functions are achieved by sequences of sub(micro)-functions (called functional achievement relations. We also call such hierarchy functional hierarchy). The next section discusses this relation in detail.

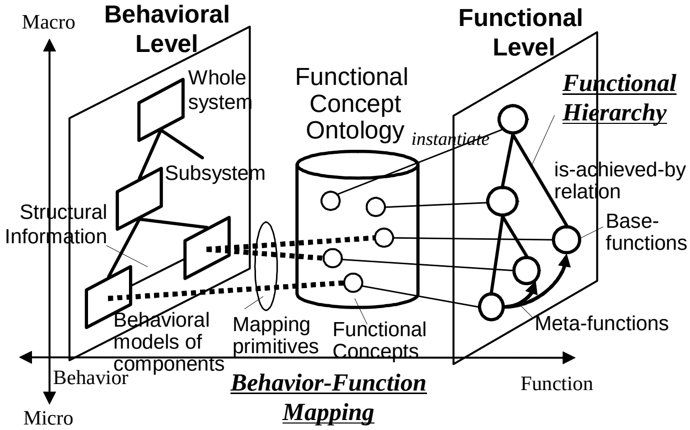

Figure 2 - Framework of functional models of artifacts

Ways of Function Achievement

Figure 3 shows a top-most part of a functional hierarchy of a power plant as an example. The required function "to generate electricity" is decomposed into "to generate torque" and "to generate electricity". The former sub-function is further decomposed into more micro-functions including "to vaporize water" and "to heat water", "to generate torque" and "to condense steam". Thus, the function achievement relations of artifacts discussed above are results of the functional decomposition in the conceptual design process.

Such a traditional functional decomposition model lacks information of the behavioral level and represents only "how" the macro-function is achieved but does not represent "why" the sequence of the sub-functions can achieve the macro-function. .

The key idea of our systematization of knowledge of functional decomposition is *way of function achievement*. When a macro-function can be achieved by a sequence of sub(micro)-functions, we call the sequence of sub-functions constrained by the relations among them *a method of the achievement*. On the other hand, we call the background knowledge of functional decomposition such as physical principles, theories, phenomena, and structure as the basis of the achievement *a way of the achievement*. For example, in Figure 3, the basis of the second functional decomposition can be represented as "thermal steam way" whose intended phenomena is (adiabatic) heat expansion of gas. Such a way of achievement of a function can represent partial information of behavioral and/or structural one.

General Knowledge of Ways of Function Achievement

In general, there are many methods (sequences of sub-function) of achievement of a function. In conceptual design, designers select an appropriate method from many alternative methods using general knowledge of ways of achievement. We call such knowledge the *ways of function achievement* as a kind of functional knowledge. In this section, its systematization is discussed.

As mentioned in Introduction, the conventional description of such knowledge tends to be scattered around technical domains, to lack consistency of viewpoints, to be ill-structured, and/or to be categorized by the non-fundamental characteristics which can be derived from deeper principles. The key points of our systematization are; (1)the concept of way, (2)common functional concepts provided by the ontology of functional concepts, and (3)categorization of the knowledge. The first one enables us to explicate the background knowledge of functional achievement and thus gives us key concepts for organization of the knowledge, which realizes consistent categorization. The second one provides us common vocabulary for representation and then makes the functional knowledge reusable. The last one enables us to distinguish

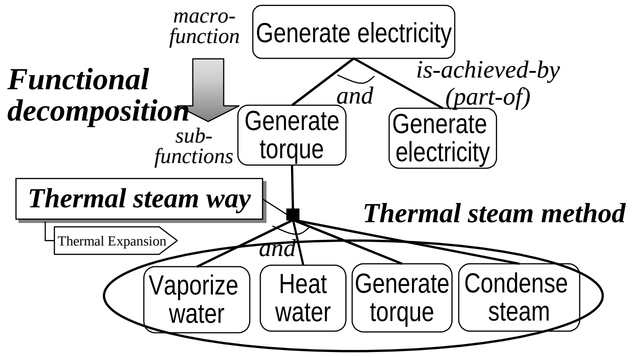

Figure 3 - A functional hierarchy of a power plant (portion)

the inherent one from derivable ones, and thus makes the knowledge well-structured. The details are discussed in the following.

Description of a Way of Function Achievement

The knowledge of achievement of a function is described as a set of ways of achievement. A description of a way of achievement of a function consists of the function as a macro-function, a set of sub(micro)-functions, temporal and causal constraints among sub-functions, principle of the achievement, conditions for use of the way, and characteristics of products using the way.

The macro- and micro-functions are described in terms of general functional concepts which are provided by the ontology of functional concepts [Kitamura 00]. Each of the provided functional concepts has general meaning of a base-function independent of target objects and components. Thus, the knowledge in terms of them can be applicable to other domains.

The principle of achievement represents physical principle or physical phenomena in order to achieve the macro-function. It governs the characteristics of the product using the way. For example, in the chemical way for connection, the chemical cohesiveness determines that the product cannot be disassembled without disrupt. The characteristics are described in qualitative manner.

Categorization of Knowledge of Function Achievement

We categorize typical representations of the knowledge into an inherent knowledge based on their principles and the other representations derivable from the inherent one as follows. Figure 4 shows the categorization and their examples.

The is-a Hierarchy of Ways

The ways of achievement of a function are organized as an is-a hierarchy according to their principles. Because the principles are inherent properties of ways, we can recognize it as a right is-a hierarchy. For example, in Figure 4, ways of connection are categorized into mechanical, chemical, and physical ways according to the principles for constraints of positions followed by further specialization. The other kinds of knowledge can be derived by reorganizing this is-a hierarchy. Note that this is-a hierarchy represents abstraction of principles in order to achieve each macro-function, that is, key information about how to achieve the function. On the other hand, the is-a hierarchy of functional concepts [Kitamura 00] represents abstraction of functions themselves, that is, what goal is achieved.

Attributes Tree

For selection of ways, ways are classified according to their attributes in a form of so-called decision tree. Each leaf node represents a way. Its structure depends on the viewpoints. In Figure 4, an attribute tree from the viewpoint of disassembly is shown. From the viewpoint, the important characteristics of a product using a specific way of connection include whether the product can be disassembled without disrupt or not and whether it can be done without deformation or not. Such characteristics appear as branch nodes in the tree. Each of the ways of connection is classified by values of such characteristics and appears as a leaf node. This type of trees is sometimes confused with *is-a* hierarchy. They are the same in the sense that both are classification of ways, but are different from each other in that while the structure of the decision tree can be changed according to the purpose of the classification of interests and the intermediate nodes do not have to correspond to meaningful and reasonable categories of ways, the structure of is-a hierarchy is determined by its inherent property and hence it is unique. Confusion of this difference has been one of the causes of the inappropriate organization of way knowledge we have used to date.

Functional Decomposition Tree (Functional Hierarchy)

This is a kind of a product model at the functional level like one shown in Figure 2. It consists of sub-functions and description of the way as shown in Figure 2. This tree is constructed as a result of decision of the product designer.

General Functional Decomposition Tree

It consists of all possible ways of achievement of a function in *OR* relationship. It is a kind of functional decomposition trees independent of a specific device. All the OR branches in a path from the root to a leaf node in a tree of this type collectively define a *composite way* which is combination of some primitive ways defined in the *is-a* hierarchy as shown in Figure 4. This type of tree is not stored in the system as it is. Like attribute trees, it is also generated by connecting each piece of functional concepts organized in an *is-a* tree upon request. In general, the ways described in textbooks are often composite one. For example, “the arc welding way” shown in a textbook is a composite of two primitive ways; “the fusion welding way of connection” and “the arc way of fusion” as shown in Figure 4. Note that the second way is not the way of connection but of fusion which is a micro-function of connection. These composite ways often cause an inappropriate structure of conventional organization of way knowledge. In our framework, they can be properly described as such composite ways in the general functional decomposition trees.

Manufacturing Activity Decomposition Tree

It consists of manufacturing activities (process) of a product. This tree of a product and the functional decomposition tree of the product are usually different from each other. In the case of static functions such as connection, however, the principles of the ways in the both trees are often the same. Note that manufacturing activities of a product can be recognized as functions of a manufacturing device for the product. Thus, the common functional concepts can be used in the both trees.

General Manufacturing Activity Decomposition Tree

It consists of possible ways of manufacturing of a product in OR relationship.

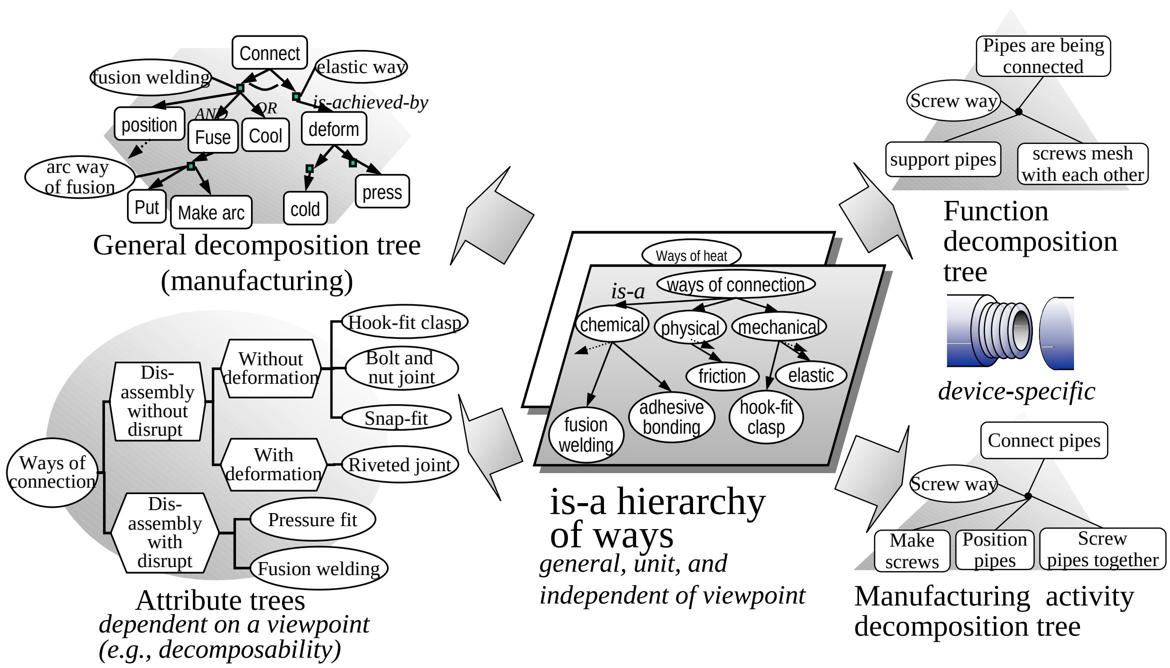

Figure 4 - Categories of knowledge of ways and their examples

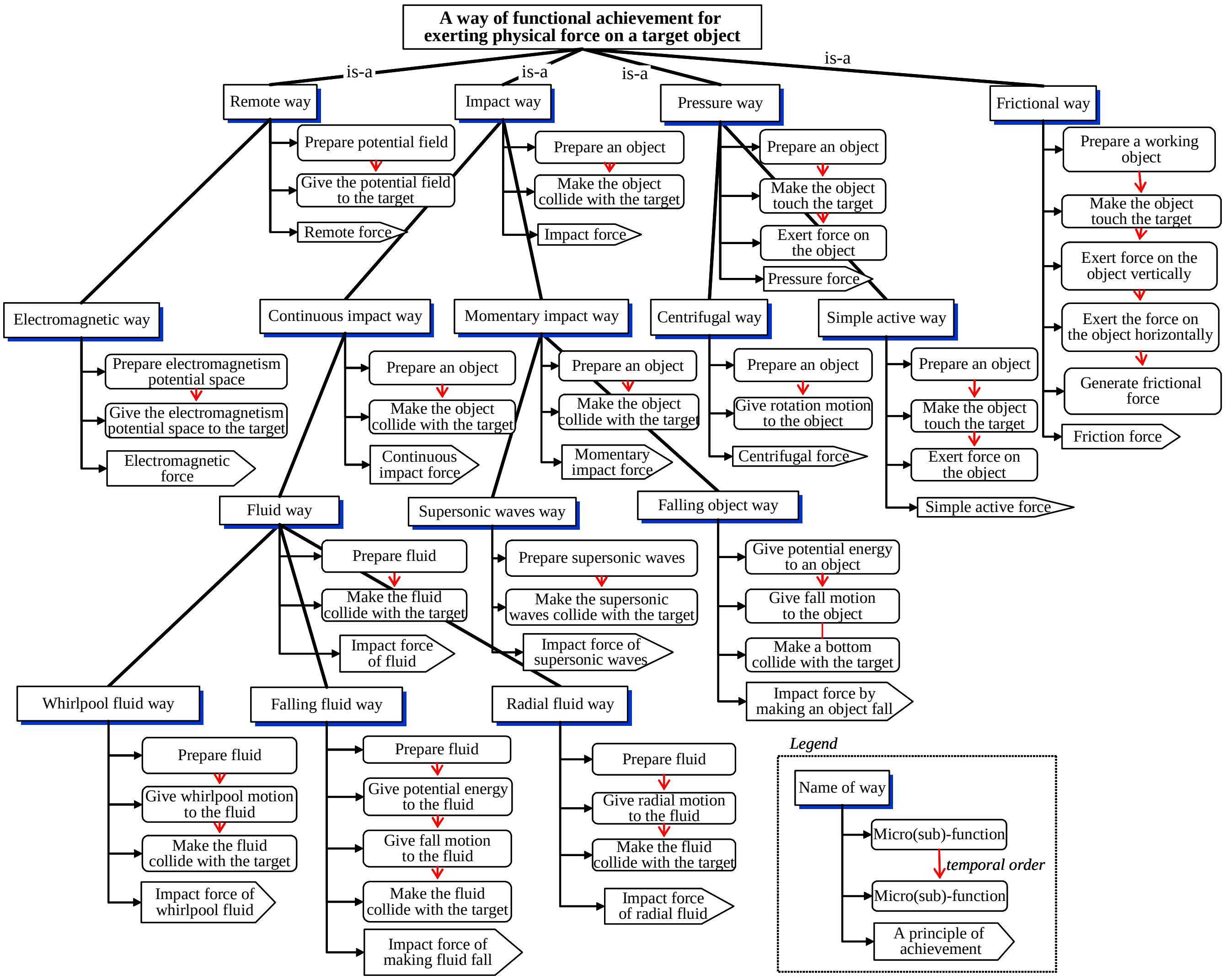

Figure 5 - The knowledge base of ways of achievement for "to exert physical force" (is-a hierarchy)

Building Knowledge Base of Ways of Function Achievement

We have described 104 ways of achievement of 26 functions from five examples; a washing machine, a printing device, slicing machines for ingot of semiconductors (using wire or rotating blade), and an etching device. Firstly, we described functional decomposition trees of them. Next, we generalized the ways appeared in the examples. Then, we tried to find their principles and then organized them into the is-a hierarchies as discussed above. Lastly, we added other ways based on principles.

As an example, Figure 5 shows an is-a hierarchy of ways of achievement for "exerting physical force". In the figure, a box, a round box, and a pentagon represent a concept of way, a sub(micro)-function of the macro-function in the way, and a principle of the way, respectively. The ways for exerting physical force are categorized into four types according to types of force; *remote force*, *impact force*, *pressure force*, and *friction force*. The *impact way* is furthermore categorized into sub-types: the *continuous impact way* and the *momentary impact way* according to length of the time interval of application of force.

All the sub-functions shown in Figure 5 can be further decomposed into finer sub-functions according to other *is-a* hierarchies of ways of achievement for each sub-function in the same manner. The results of decomposition can be used to form a functional decomposition tree or a general functional decomposition tree as discussed in the previous section. Thus, is-a hierarchies of ways of achievement can generate "is-achieved-by" relations (a kind of part-of relations) among functions.

The knowledge of ways of achievement is general and then applicable to many systems. In fact, the ways shown in Figure 5 are commonly used in some of the five examples. For example, many of washing machines use the *fluid way* for exerting force to cloth, while in the slicing machine the same way is also used in order to remove cutting dust.

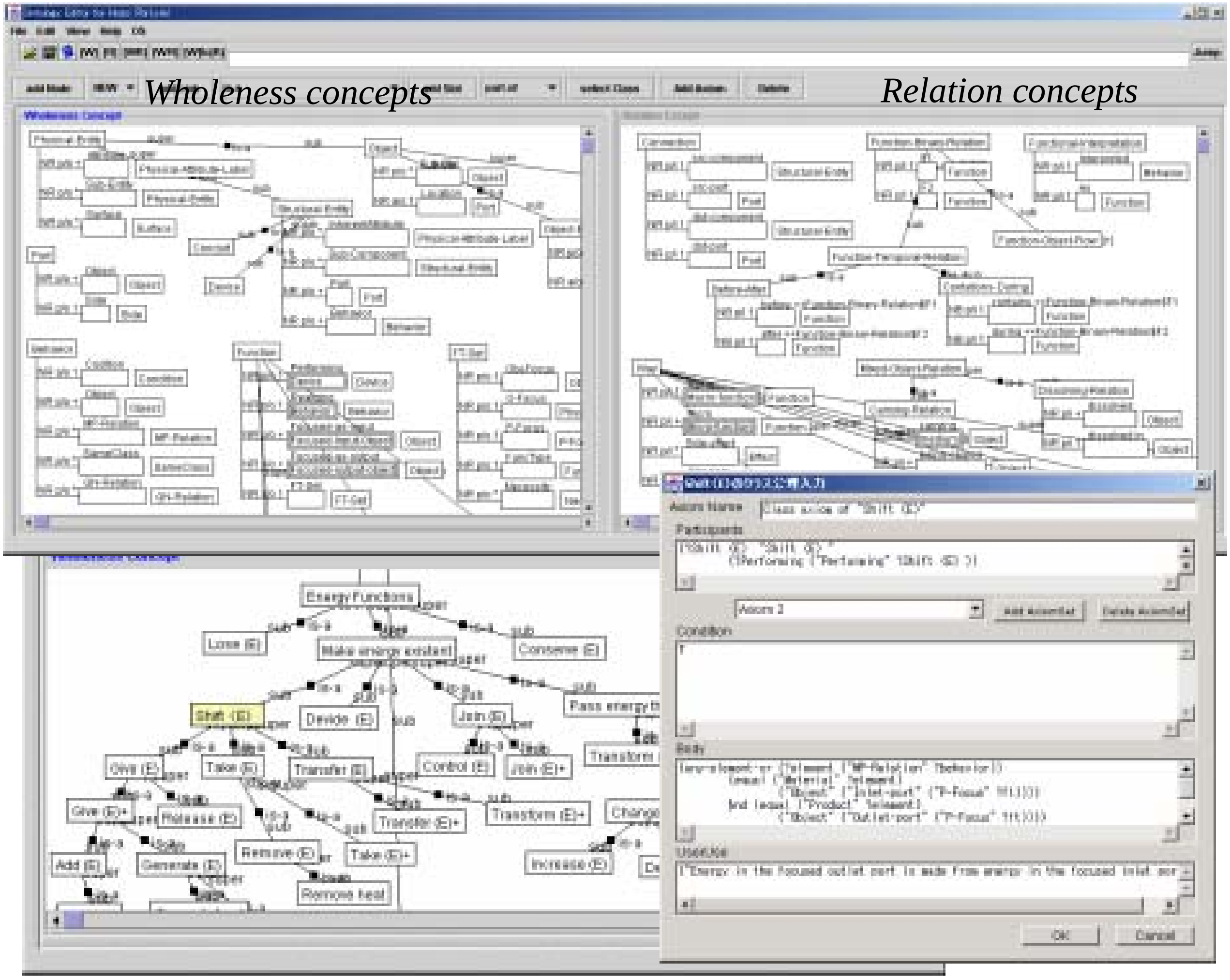

Figure 6 - The implemented device ontology (the above) and the functional concept ontology (the below) (portion)

Ontology-based Environment for Description of Functional Knowledge

The framework of description of functional knowledge discussed thus far has been implemented using our environment for building and using ontologies named Hozo [Kozaki 00]. The ontology editor of Hozo helps ontology authors define concepts in ontologies as class-frames and describe instance models based on the ontologies by instantiating the classes. In this environment, a class-frame of a concept consists a label of the concept, a super-concept in the is-a hierarchy, slots of part concepts (part-of relation denoted by p/o), slots of attributes (a/o), and axioms. Definition of each slot consists of a class constraint for slot values (instances) and a role of the slot values in the frame's context. Concepts are categorized into the wholeness concepts composed by the part concepts and the relation concepts between the concepts. For more details, see other article [Kozaki 00].

We implemented our functional ontologies using the ontology editor of Hozo and then can use Hozo as an environment for description of functional knowledge and functional models. Note that Hozo is not specific to the functional ontologies but a general tool for ontologies. Thus, although its user interface is not helpful for naive authors who are not familiar with ontologies, it shows fundamental effects of functional ontologies for description of functional knowledge.

Implementation of Functional Ontologies

Figure 6 shows a portion of the implemented functional ontologies in Hozo. The upper window shows definitions of the device ontology as a modeling language of function and behavior named FBRL (Function Behavior Representation Language) [Sasajima 95]. It includes the class-frames of device, behavior, function, and functional toppings (FTs). For example, a device is a sub-concept of a structural entity and consists of physical attributes, sub(part)-components representing whole-part relations among devices, ports for connection with other devices, and its behaviors. The FTs represent information of teleological interpretation of a behavior and then represent mapping from the behavior to a functional concept. A set of FTs is composed of four items (as denoted by p/o slots in the figure):

(1) *Obj-Focus* specifies *objects* to focus on

(2) *O-Focus* specifies attributes of the *object* to focus on
(3) *P-Focus* specifies port to focus on
(4) *Necessity* specifies the necessity of *objects*

They enable us to define functional concepts explicitly in machine understandable forms as follows.

The lower window in Figure 6 shows a portion of the functional concept ontology [Kitamura 99, 00]. It consists of base-functions for functions of devices, function types, and meta-functions representing collaborative roles in interdependency among base-functions. The base functions are categorized by kinds of target objects such as substance, energy, information, force and motion. The lower window in Figure 6 shows a portion of an is-a hierarchy of the energy-related base-functions.

Each concept in the hierarchies is defined using FBRL and hence operational. For example, an energy function, *To shift energy*, is defined as a behavioral constraint: *focused energy moves between two different mediums.* It can be defined by the following three axioms;

- P-Focus on an inlet port and an outlet port.
- Energy in the focused outlet port is made from energy in the focused inlet port.
- Mediums of the focused energies are different.

The right window in Figure 6 shows the window for describing the second axiom. The "participant" field defines arguments (parameters) appeared in the axiom body, "axiom body" defines constraints over arguments. The constraint should be true for all instances of the class if the "condition" is true (the symbol "T" means "always true"). The "UserUse" field can be used for comments.

To take, a sub-function of *to shift* in the is-a hierarchy, is defined as that of *to shift* with an additional FT, P-Focus on the port of energy provider. Moreover, *To remove* is defined as that of *to take* with an additional FT, the energy taken is *unnecessary* as *Necessity* FT.

These definitions demonstrate high independence of their implementation, while function is clearly related to structure and behavior.

Description of Ways

A way of functional achievement is defined as a subclass of the way relation-concept. The class of "way relation" is defined in the device ontology shown in the upper window of Figure 6. It consists of a macro-function, micro-functions, principle for the functional achievement and side-effects.

Figure 7 shows an example of the implemented ways. It shows an implementation of the frictional way for exerting physical force in Figure 4. Its macro-function is "to exert force" and its sub-functions including "to make an object tough another object" and "to exert force to the objects". There are (should be) relations among sub-functions such as temporal relation (in the figure, Contains-During) and meta-functions (ToProvide).

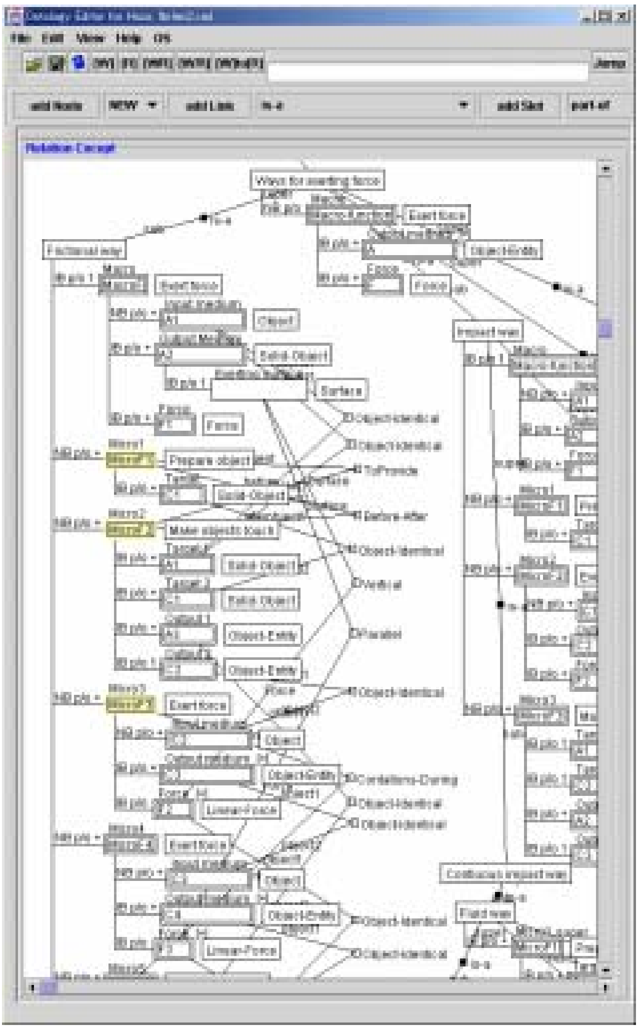

Figure 7 - An example of implemented ways

Functional Way Server

This section shows a design supporting system named a functional way server as an application of the functional knowledge described based on the ontologies. The server is designed to support conceptual design of engineering devices, providing suitable ways of achievement of the function that designers consider. Figure 8 shows its framework and examples of provided knowledge. The server contains the knowledge of ways of function achievement of as is-a hierarchies. Given a required function and a viewpoint of the designer, the server reorganizes the knowledge in a manner suitable for the viewpoint and then shows ways in the form of an attribute (decision) tree. A viewpoint represents a context of designer's thinking process. It consists of a phase in the product lifecycle such as manufacturing, using (product design), and recycling, focused attribute set, grain size, and a domain of interest such as mechanical and electrical.

Here we show a possible scenario of design using the functional knowledge server (see Figure 8). Let us suppose a concurrent design team are designing a connection between an air-conditioner and a pipe. Firstly, a product designer A is selecting a way of connection from the

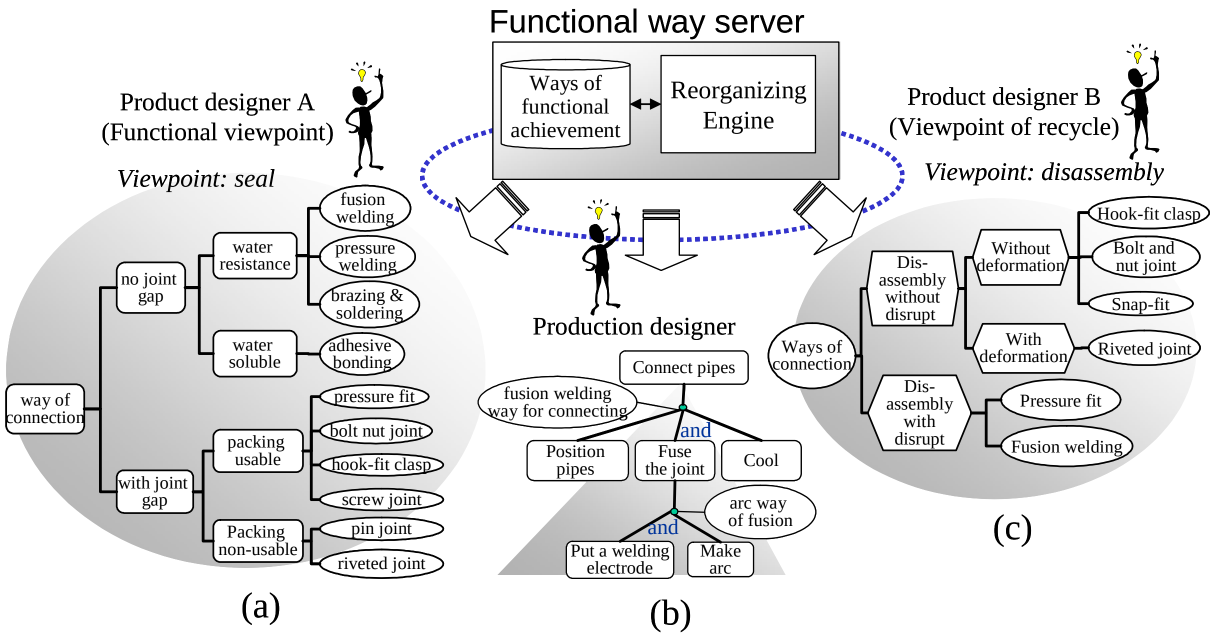

Figure 8 - Framework of the functional way server and a scenario

viewpoints of seal in order to seal the coolant in the pipe. Given the viewpoint “seal”, the server looks up suitable attributes such as “joint gap” and “water resistance” associated with the viewpoint. According to the values of such attributes, the server classifies suitable ways, then shows the designer A an attribute-tree shown in Figure 8a. If the product designer would select the “fusion welding” way, a functional decomposition tree and a manufacturing activity decomposition tree are constructed (Figure 8b). A manufacturing designer can decompose the manufacturing-activity by selecting a way of fusion, say, the arc way.

Next, suppose that other designer B from the viewpoint of recycling checks the connection. The server derives the attribute tree shown in Figure 8c from the viewpoint of disassembly using the same knowledge of ways. If the designer B would submit an alternative plan “to hook-fit clasp”, the product designer A would recognize the characteristics of “hook-fit clasp” on seal in Figure 8a and thus add a sealant in order to seal the coolant.

The server provides a wide range of ways in different domains and then facilitates innovative design. Moreover, because the server can track effects of decisions made by both the product designer and the manufacturing designer, it facilitates negotiation in a concurrent design team.

Related Work and Discussion

In a literature on design, many general knowledge for functional decomposition or functional synthesis are proposed (e.g., [Bradshaw 91; Umeda 96]). The major differences between our ways of achievement and them include explicit description of “way”, organization in is-a hierarchies based on principles of ways, and an ontology of functional concepts as follows.

Firstly, our ways of functional achievement explicitly represent the feature of achievement such as theory and phenomena. They enable designers to facilitate the smooth interaction between the structural level and the functional level. The designer can check the feasibility of functional decomposition using the features represented as the ways as behavioral constraint. Such functional decomposition is compliant with the observations found in the research on design processes [Takeda 90] in which they say functional decomposition is not done solely in the functional space but done by going back and forth between the functional, behavioral and structural spaces during which some portions of the artifact are determined in each of the spaces simultaneously.

Secondly, we organized such general knowledge as an is-a hierarchy. Although the feature knowledge is also captured in [Malmqvist 97], it corresponds to the functional decomposition tree of a specific product in our categorization and there is no organization of general knowledge. The crucial issue of organization as an is-a hierarchy is to capture inherent properties in order to avoid improper categorization as mentioned in Introduction. The concept of way enables us to capture the principles of achievement of a function (i.e., why the function can be achieved) as the inherent properties and thus to realize consistent categorization.

Lastly, our functional knowledge is based on a functional concept ontology [Kitamura 00]. As mentioned in Introduction, there is no rich and generic vocabulary of functional concepts. Thus, the conventional functional concepts are rather ad-hoc and/or dependent on specific implementation of the component. This is why we often see inconsistent descriptions of functional concepts and cases where different labels and definitions are given to a single function. We have identified about one hundred generic functional concepts with clear operational definitions in is-a hierarchies as discussed with Figure 6. Use of such functional concepts as vocabulary of description of ways of achievement facilitates reuse of the knowledge in different domains.

Such systematization of functional knowledge based on the concept of way and the functional concepts enables the design supporting system to provide a wide range of alternative ways in different domains. Such knowledge in different domain can facilitate innovative design many of which are based on techniques well-known in different domains [Sushkov 95]. TechOptimizer [IMC 99] is a software product based on a theory for innovative design [Sushkov 95], which contains generic principles of invention. It, however, just searches highly abstract principles for given criteria. It is not adaptive for the designers' viewpoints, while our functional way server reorganizes the general knowledge according to the designer's viewpoint.

Summary

The contribution of this research can be summarized as a framework of systematization of design knowledge about functional decomposition. The benefits of such functional knowledge in a concurrent design team were shown using a scenario.

Acknowledgments

The authors would like to thank Kouji Kozaki, Mariko Yoshikawa, Tomonobu Takahashi, Masaru Takahashi, Toshio Ueda and Kouki Higashide for their contributions. Special thanks go to Dr. Masayoshi Fuse, Sumitomo Electric Industries Ltd., for his cooperation for application to manufacturing processes. This research is supported in part by the Japan Society for the Promotion of Science (JSPS-RFTF97P00701). Part of this research was part of GNOSIS project within the Intelligent Manufacturing Systems (IMS) research program.

References

[Bradshaw 91] Bradshaw, J. A., and Young, R. M., Evaluating Design using Knowledge of Purpose and Knowledge of Structure, *IEEE Expert*, 6(2):33-40, 1991.

[Chandrasekaran 93] Chandrasekaran, B.; Goel, A. K.; and Iwasaki, Y., Functional Representation as Design Rationale, *Computer*, 48-56, 1993.

[de Kleer 84] de Kleer, J. and Brown, J. S., A Qualitative Physics Based on Confluences, *Artificial Intelligence*, 24, 7-83, 1984.

[Gruber, 93] Gruber, T. R.. A Translation Approach to Portable Ontology Specifications, *Knowledge Acquisition*, 5(2):199-220, 1993.

[Hodges 92] Hodges, J., Naive Mechanics - a Computational Model of Device Use and Function in Design Improvisation, *IEEE Expert*, 7(1):14-27, 1992.

[IMC 99] Invention Machine Co., TechOptimizer, http:// www.invention-machine.com/products/techoptimizer.cfm, 1999

[Keukene 91] Keuneke,A.M., Device Representation: the Significance of Functional Knowledge, *IEEE Expert*, 24:22-25, 1991

[Kitamura 99] Kitamura, Y., and Mizoguchi, R., Meta-functions in Artifacts, *Papers of 13th International Workshop on Qualitative Reasoning (QR-99)*, 136-145, 1999

[Kitamura 00] Kitamura, Y., Sano, T., Mizoguchi, R., Functional Understanding based on an Ontology of Functional Concepts, In *Proc. of The Sixth Pacific Rim International Conference on Artificial Intelligence (PRICAI 2000)*, pp.723-733, 2000

[Kozaki 00] Kozaki, K., Kitamura, Y., Ikeda, M., and Mizoguchi, R., Development of an Environment for Building Ontologies which is based on a Fundamental Consideration of "Relationship" and "Role", In *Proc. of The Sixth Pacific Knowledge Acquisition Workshop (PKAW 2000)*, pp.205-221 ,2000

[Lind 94] Lind, M., Modeling Goals and Functions of Complex Industrial Plant, *Applied Artificial Intelligence*, 8:259-283, 1994.

[Malmqvist 97] Malmqvist, J., Improved Function-means Trees by Inclusion of Design History Information, *Journal of Engineering Design*, 8(2):107-117, 1997.

[Mizoguchi 00] Mizoguchi, R., and Kitamura, Y., Foundation of Knowledge Systematization: Role of Ontological Engineering, *Industrial Knowledge Management - A Micro Level Approach*, Rajkumar Roy Ed., pp.17-36, Springer-Verlag, London, 2000

[Pahl 88] Pahl, G., and Beitz, W., *Engineering Design - a Systematic Approach*, The Design Council, 1988.

[Saasajima 95] Sasajima, M.; Kitamura, Y.; Ikeda, M.; and Mizoguchi, R. FBRL: A Function and Behavior Representation Language. *In Proc. of IJCAI-95*, 1830-1836, 1995.

[Sushkov 95] Sushkov, V.V. Mars, N.J.I., and Wognum, P.M., Introduction to TIPS: a Theory for Creative Design, *Artificial Intelligence in Engineering*, 9, 1995

[Takeda 90] Takeda, H. et. al., Modeling Design Processes, *AI Magazine*, 11(4), 37-48, 1990

[Tejima 81] Tejima, N. et al. eds., Selection of Functional Terms and Categorization, Report 49, Soc. of Japanese Value Engineering (In Japanese), 1981

[Umeda 96] Umeda, Y., Ishii, M., Yoshioka, M., Shimomura, Y., and Tomiyama, T., Supporting conceptual design based on the function-behavior-state modeler, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 10, 275-288, 1996.