DESIGN OF FUNCTIONS BY FUNCTION BLENDING

Yu Park, Shota Ohashi, Eiko Yamamoto and Toshiharu Taura
Kobe University, Japan

ABSTRACT

This study aims to develop a method for supporting the design of a new function. Currently, in product development and design, the question “What should we create?” is the main issue to be addressed, whereas previously, the main issue was “How should we realize the given function?” In this study, we approach the current issue by designing a new function. We propose the method of “function blending” to support the design of a new function structure. This method is systematized from a linguistic viewpoint so that a new function structure can be derived using linguistic hierarchal relations. We develop the formulation of function blending in a function dividing process, and a method for developing a thesaurus for function blending. Finally, we confirm the feasibility of the proposed method.

Keywords: Function design, function blending, thesaurus, linguistic hierarchal relation

1 INTRODUCTION

Currently, our living environment has become filled with products, and truly excellent products have become a greater requirement for consumers than ever before. In this situation, products have come to be evaluated by consumers on the basis of their functionality. Therefore, functional originality is required in product development and design. “What should we create?” is now becoming the main issue to be discussed, whereas in the past, it was “How should we realize the given function?” We identify “What” as the specification or function. Accordingly, in conceptual design, the focus should be on not only deriving a novel mechanism but also creating a novel function.

In traditional conceptual design, function description has been used to represent the required specifications, and various function-description-based models of the design process have been developed, aimed at design support [1], [2], [3], [4], [5]. However, because the aim of these methods is to answer the question “How should we realize the given function?”, it might be difficult to answer the question “What should we create?” We think the difficulty also comes from the perspective that the issue of “What should we create?” cannot be discussed within the existing framework of problem solving [6].

It is necessary to create (design) the actual specification (function) to answer the question “What should we create?” Therefore, it is necessary to investigate the functions from the design viewpoint. On the abovementioned basis, in this study, we consider the question of “What should we create?” as equivalent to “designing a new function,” and propose an associated methodology. What should we do to design a function? The human approach to creating a new concept is often illogical and experimental. However, some rules are expected to exist in the thinking process, and these rules are thought to be effective in designing a function. For example, concept blending has been recognized as an effective method to create new concepts in the field of design research and cognitive science research [7]. This method is used to develop a truly new concept by blending two base concepts; that is, the developed concept does not belong to the domain of either of its two base concepts, though it partially incorporates the features of each base concept. It has been mentioned that this method is effective in generating a highly creative concept. In this study, by focusing on the notion of concept blending, we propose a methodology that generates the new function from some base functions and attempt to support the design of a function. Concretely, as a process for designing a function, we blend some functions to derive a new lower-level function structure, which is original and feasible. We focus on the novelty of function structures because the new function becomes meaningful only when the function can be implemented into a concrete mechanism. This is because the design of a function is different from idea generation, which does not require the implementation of a mechanism. To support this function design, it is necessary to elaborate on current functional decomposition processes in the field of engineering design to blend functions and develop a methodology for mapping between upper-and lower-level functions. In our previous study, a method was developed that can be implemented on a computer, supporting the functional decomposition process. This method identifies the hierarchal relations between upper- and lower-level functions in the functional decomposition process, from the viewpoint of linguistic hierarchal relations between words describing the upper- and lower-level functions [8]. Furthermore, we have developed a thesaurus, which supports the mapping between the upper- and lower-level functions. In this study, we expand this method and propose a methodology that can support the design of a new function structure by blending some functions. For this, we first conduct a case study on the basis of the notion of concept blending and find out how to blend functions. That is, we identify operating types for function blending. Next, we formulate the process of function blending of the identified operating types. We simultaneously develop a thesaurus that can support the procedure of function blending. Finally, we develop a computer system that implements the function blending process.

2 FUNCTION BLENDING IN FUNCTION DIVIDING PROCESS

Functions are usually expressed by using words, for example, "washing dishes." We have proposed a method for mapping the hierarchal relations between the upper- and lower-level functions by using the linguistic hierarchal relations between the words on the upper- and lower-level functions [8]. We developed a method for obtaining lower-level functions from upper-level functions (hereafter called the "function dividing process" (FDP)). In the study, we first conducted a case study and found the operating types in the FDP. Next, we systemized the identified operating types from the viewpoint of linguistic hierarchal relations between the words on the upper- and lower-level functions. Figure 1 shows the relationship between the FDPs and the function operating types. We then constructed a thesaurus that can support the FDP. In our formulation of the FDP, there exist two types of divisions: the "decomposition-based dividing process" and the "causal-connection-based dividing process."

The causal-connection-based dividing process deals with the causal effects of physical phenomena among the components (see the right side of Figure 1). Two functions are said to be causally connected when an object of one function is equal to a subject of the adjacent function at the same level. In the causal-connection-based dividing process, a chain of causal relations connects the lower-level functions; an object of the terminal lower-level function is equal to that of the upper-level function. In addition, the verb of the terminal lower-level function is equal to that of the upper-level function. In the decomposition-based dividing process, the subset of subjects of the lower-level functions is a component of the subject of the upper-level function; the subset of verbs of the lower-level functions is the action that realizes the verb of the upper-level function. The object of each lower-level function is equal to that of each upper-level function (see the left side of Figure 1).

Here, a function is defined as "an entity's behavior that plays a special role" and is described using a subject (*S*), a verb (*V*), and an object (*O*). We identify a subject as an entity having a certain role. This identification makes it possible to explain the structure of engineering products. On the basis of the abovementioned previous study, in this study, we develop a method to construct a new function structure by blending some existing functions (hereafter called "function blending in the FDP").

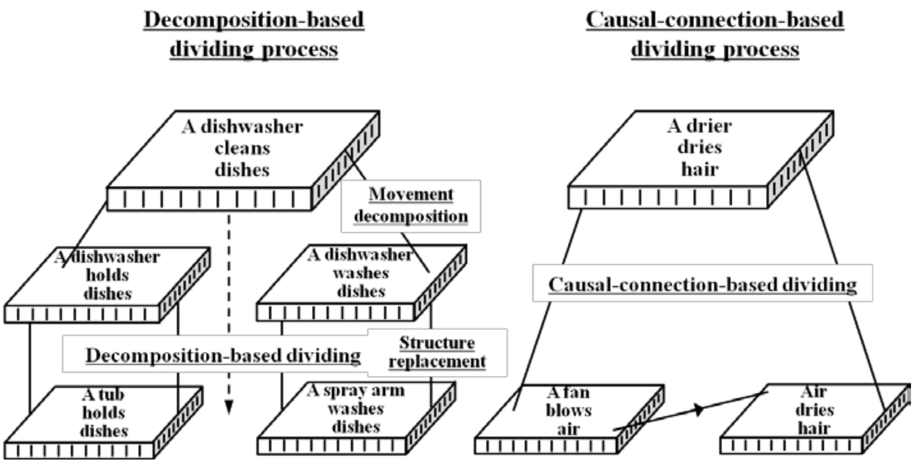

Figure 1. Relationship between function dividing process (FDP) and function operating types

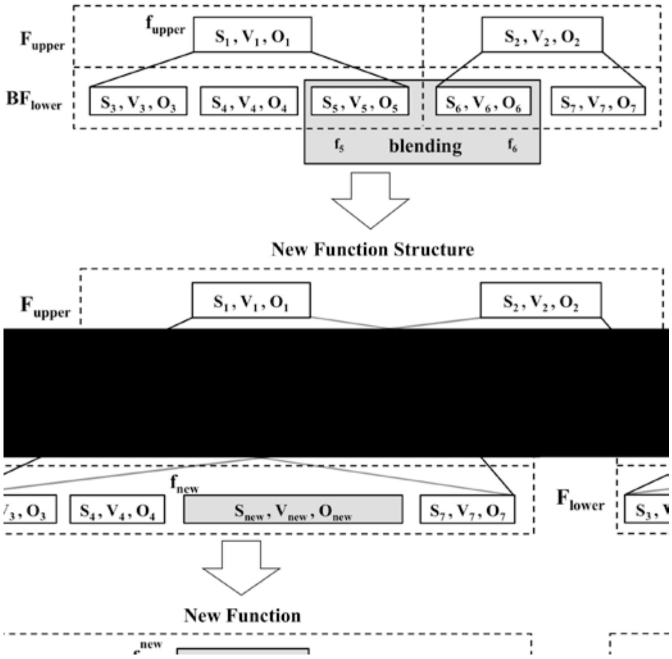

Figure 2. Process of designing function

2.1 Defining Function Blending in the FDP

In this study, we take the notion of “concept blending” [7], which is a focal method used to create a new concept in design research and cognitive science research, in the design of a function. First, we define the blending of some existing functions, to design a new function, as the “function blending operation” (FBO). Next, we define “obtaining a new set of lower-level functions from an existing set of lower-level functions,” via the FBO, as “function blending in the FDP.” We can define function blending in the FDP as follows:

$$m : F_{upper} \rightarrow F_{lower} \tag{1}$$

where F_{upper} denotes the set of upper-level functions and F_{lower} denotes the set of new lower-level functions. Figure 2 depicts the flow of the process of designing a new function, which involves function blending in the FDP. In this figure, BF_{lower} denotes the set of lower-level functions, which are obtained by dividing the upper-level functions. First, a new lower-level function structure is obtained via the FBO (upper figure -> middle figure in Figure 2). Next, a new upper-level function is obtained from the newly obtained lower-level function structure (middle figure -> lower figure in Figure 2). In this paper, we discuss the process used to obtain a new lower-level function structure via the FBO (upper figure -> middle figure in Figure 2).

2.2 Classifying Function Blending in the FDP

We conducted a case study to formulate the process of function blending in the FDP. We analyzed 13 examples of engineering products, which can be viewed as the result of blending the functions of two existing products. A “flashlight with hand generator” is an example. It has a “flashlight” function and a “hand generator” function. The case study results revealed that the FBOs can be classified into two types: “integration operation” and “conversion operation.” We describe the formulation of these operations in more detail later. An integration operation integrates two functions. For example, by integrating the digital camera function of “display shows the visual image” and the projector function of “optical unit projects the visual image” into a new function “{display and optical unit} {shows and projects} the visual image,” a new engineering product “a digital camera with projector” can be designed. On the other hand, a conversion operation replaces an existing function with another function. For example, by replacing the flashlight function “a battery generates electricity” with the hand generator function “a hand generator generates electricity,” the flashlight with hand generator function “a hand generator generates electricity” can be designed.

On the basis of these analyses, function blending in the FDPs can be classified into “function blending in the FDP by integration” and “function blending in the FDP by conversion.”

In this study, we describe a function, f = (S, V, O), by using a subject (S), a verb (V), and an object (O). The subject (S) is assumed to represent a mechanism to achieve a verb (V) and an object (O). To capture a function hierarchy, Pahl and Beitz proposed a function structure diagram in which functions are regarded as input/output relations represented by nouns and verbs [9]. Miles proposed a method of defining a function in terms of a linguistic expression for value engineering [10]. In these methods, verbs and objects constitute a type of linguistic expression. By taking these considerations into account, in this study, a function is constituted by "a subject" and "a verb and an object." Accordingly, the FBOs are classified into "to obtain a new subject" and "to obtain both the new verb and the new object" or both. Table 1 lists the classification of function blending.

Table 1. Classification of function blending operations

	Integration	Conversion
New subject	Function blending in the FDP by subject integration	Function blending in the FDP by subject conversion
New verb and/or object	Function blending in the FDP by verb or object integration	Function blending in the FDP by verb and object conversion

2.3 Defining Word Abstraction-concretion Operation

To conduct the integration and conversion operations, it is necessary to determine whether two functions are similar (that they at least overlap partially). Therefore, we introduce an operation that abstracts or concretizes the words described in the functions and term it the "word abstraction-concretion operation." This operation consists of an abstraction operation and a concretion operation. The abstraction operation obtains an abstract word from a given word, by extracting its common and/or essential nature. An example is to obtain "food" from "edible fruit." The concretion operation obtains a more concrete word from the given word by making the content of the given word clear. An example is to obtain "an apple" from "edible fruit." We define a set of words, $P(w)$, obtained by the abstraction-concretion operation for a word w as follows:

$$P(w) = \{w_i \mid w_i \text{ is a word obtained by abstracting or concreting } w\} \tag{2}$$

Here, suppose w_1 is included in $P(w_2)$, that is, $w_1 \in P(w_2)$. In this case, when both w_1 and w_2 are nouns, we define the relation between them as a "superordinate-subordinate relation of nouns." Similarly, when both w_1 and w_2 are verbs, we define the relation between them as a "superordinate-subordinate relation of verbs." These relations are well known as hierarchal semantic relations. Therefore, we can identify them in existing dictionaries and thesauri.

2.4 Defining Word Match

We define the situation in which two words are found to be similar through the word abstraction-concretion operation as a "word match." A set, $I(w)$, of words that match another word w is defined as follows:

$$I(w) = \{w_i \mid w_i = w \lor w_i \in P(w) \lor 0 < |P(w) \cap P(w_i)|\} \tag{3}$$

A set, $M(w_1, w_2)$, of words where the two words match each other is defined as follows:

$$M(w_1, w_2) = \{w_i \mid w_i = w_1 \in P(w_2) \lor w_i = w_2 \in P(w_1) \lor w_i \in P(w_1) \cap P(w_2) \lor w_i = w_1 = w_2\} \tag{4}$$

2.5 Formulating Function Blending in the FDP

By introducing the word abstraction-concretion operation and the concept of a word match, the similarity among verbs or nouns can be determined. In this section, by using these operations, we formulate the function blending in the FDP, which is implemented by integration or conversion.

Function Blending in the FDP by Integration

As listed in Table 1, integration operations can be classified into two operating types. When a word match exists between the verbs of two functions as well as the objects, we define the operation of integrating the two functions by joining their subjects as "function blending in the FDP by subject integration." When a word match exists between the subjects of two functions as well as between the verbs or objects of the two functions, we define the operation of integrating the two functions by joining their verbs or objects as "function blending in the FDP by verb or object integration." We describe the operation of joining two words using the "+" operator. Using this "+" operator, we formulate these operations as follows:

(1) Function blending in the FDP by subject integration (see Figure 3)

$$
\begin{aligned}
& m_5 : F_{upper} \rightarrow F_{lower} \\
& f_1 \in BF_{lower},\ f_1 = (S_1, V_1, O_1),\ f_2 \in BF_{lower},\ f_2 = (S_2, V_2, O_2) \\
& when\ \ V \in M(V_1, V_2),\ O \in M(O_1, O_2) \\
& f_{new} = (S_1 + S_2, V, O),\ F_{lower} = \{f_i \mid f_i \in (BF_{lower} - \{f_1\} - \{f_2\}) \cup \{f_{new}\}\}
\end{aligned} \tag{5}
$$

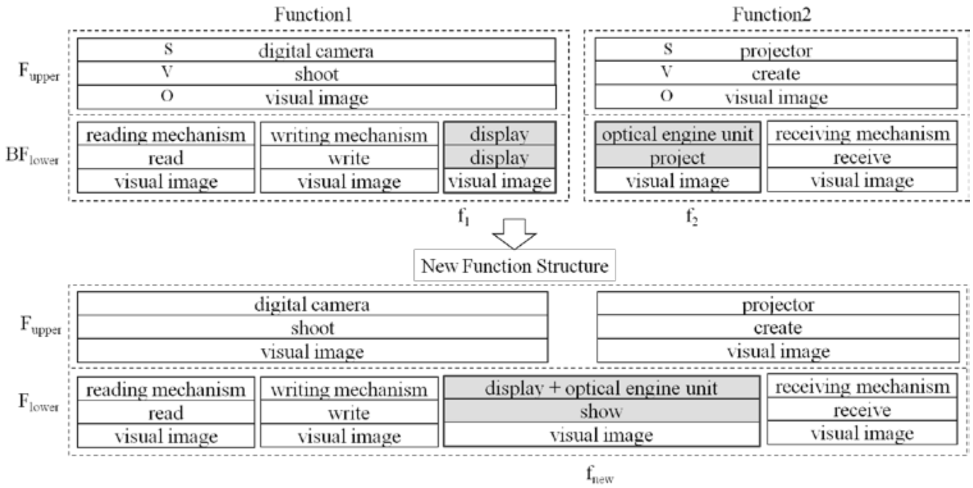

Figure 3. Example of function blending in the FDP by subject integration

(2) Function blending in the FDP by verb or object integration (see Figure 4)

$$
\begin{aligned}
& m_6 : F_{upper} \rightarrow F_{lower} \\
& f_1 \in BF_{lower},\ f_1 = (S_1, V_1, O_1),\ f_2 \in BF_{lower},\ f_2 = (S_2, V_2, O_2) \\
& when\ \ S \in M(S_1, S_2),\ V \in M(V_1, V_2),\ O \in M(O_1, O_2) \\
& f_{new} = (S, V_1 + V_2, O) \vee (S, V, O_1 + O_2),\ F_{lower} = \{f_i \mid f_i \in (BF_{lower} - \{f_1\} - \{f_2\}) \cup \{f_{new}\}\}
\end{aligned} \tag{6}
$$

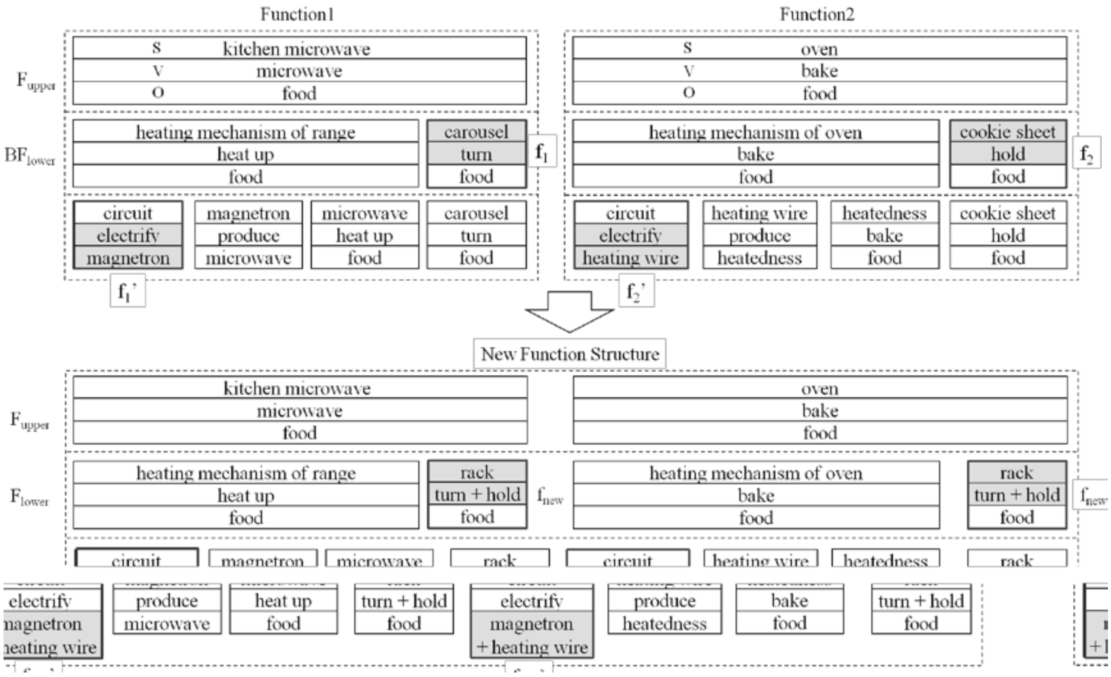

Figure 4. Example of function blending in the FDP by verb or object integration

Function Blending in the FDP by Conversion
Similarly, conversion operations can be classified into two operating types. We define the operation of replacing a function with another function, when a word match exists between the verbs and objects of the two functions, as a "subject conversion operation." An example of this conversion operation is the "flashlight with hand generator" design, discussed in section 2.2. We formulate function blending in the FDP by subject conversion as follows:

(3) Function blending in the FDP by subject conversion (see Figure 5)

$$\begin{aligned}
&m_7 : F_{upper} \rightarrow F_{lower} \\
&f_1 \in BF_{lower},\ f_1 = (S_1, V_1, O_1),\ f_2 = (S_2, V_2, O_2) \\
&when\ \ V_1 \in I(V_2),\ O_1 \in I(O_2) \\
&f_{new} = (S_2, V_1, O_1),\ F_{lower} = \{f_i \mid f_i \in (BF_{lower} - \{f_1\}) \cup \{f_{new}\}\}
\end{aligned} \tag{7}$$

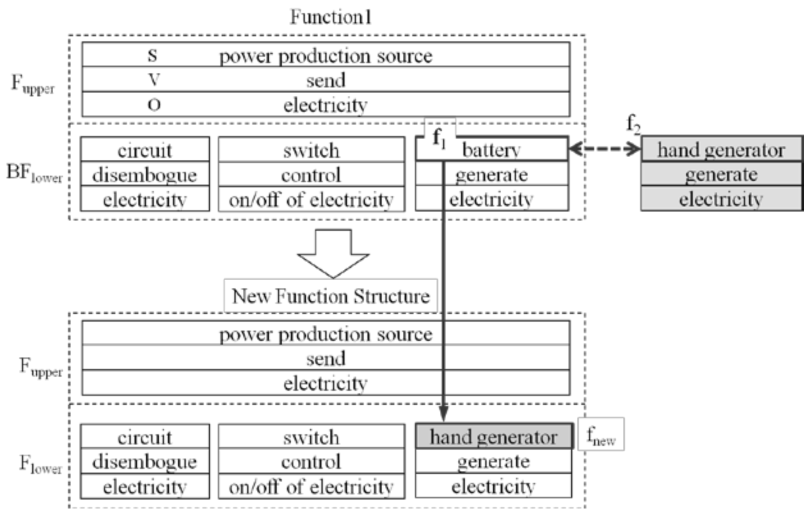

Figure 5. Example of function blending in the FDP by subject conversion

On the other hand, the verb and object conversion operation can be viewed as an operation to capture the role that a component plays from a different perspective. For example, the function of "fan blows air" can be captured instead as a function of "fan cools something," simply by changing one's viewpoint. Although engineering products manifest several functions satisfying the requirements provided by a designer, some other functions can be observed under different circumstances. For example, chairs are designed to provide a seat for a human; however, they may also function as stepladders. A function satisfying the requirements provided by a designer can be defined as a "visible function," whereas different functions that manifest under different circumstances are defined as "latent functions" [11]. A latent function can be understood as another function that is discovered when the product is observed from a different viewpoint. This action is nothing more than obtaining a new function, f_{new} = (S_1, V_2, O_2), by replacing a verb and an object of some other existing function, from f_1 = (S_1, V_1, O_1), as shown in Figure 6. From this understanding, a method of reasoning out latent functions based on some visible function has been proposed [12]. The method entails inferring a latent function by transferring a function from another engineering product to the visible function of the focal engineering product. Concretely, when there is a commonality amongst the lower-level functions of these two products, a latent function, f_1^{new} = (S_1, V_2, O_2), can be obtained by replacing verb V_1 and object O_1, from the function f_1 = (S_1, V_1, O_1) of the product, with verb V_2 and object O_2 of a function, f_2 = (S_2, V_2, O_2), of another product. This operation is the same as the verb and object conversion operation outlined in this study.
On the basis of the above considerations, we formulate "function blending in the FDP by verb and object conversion." We formulate this process to address two cases: when a causal connection is included in both of the lower functions, and when it is not.

(4-1) Function blending in the FDP by verb and object conversion, when a causal connection is not included in both F_{lower}^{f1} and F_{lower}^{f2} (see Figure 6). This case is valid only when there are more than N common functions between F_{lower}^{f1} and F_{lower}^{f2} (i.e., $| F_{lower}^{f1} \cap F_{lower}^{f2} | \geq N$ (constant)).

$$
\begin{aligned}
&m_8 : F_{upper} \rightarrow F_{lower} \\
&f_1 \in BF_{lower}\,,\; f_1 = (S_1, V_1, O_1)\,,\; f_2 = (S_2, V_2, O_2) \\
&when\; \left| F_{lower}{}^{f1} \cap F_{lower}{}^{f2} \right| \geq N\;(constant) \\
&f_{new} = (S_1, V_2, O_2)\,,\; F_{lower} = \{ f_i \mid f_i \in (BF_{lower} - \{f_1\}) \cup \{f_{new}\} \}
\end{aligned} \tag{8}
$$

Here, f_1 denotes a lower-level function of some upper function and f_2 denotes a lower-level function of some other upper function. In addition, $F_{lower}{}^{f1}$ denotes the set of lower-level functions when the upper-level function is f_1, and $F_{lower}{}^{f2}$ denotes the set of lower-level functions when the upper-level function is f_2.

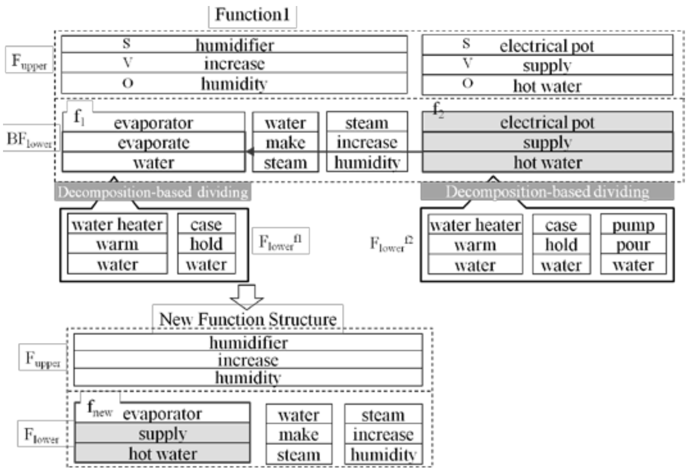

Figure 6. Example of function blending in the FDP by verb and object conversion, based on decomposition-based dividing (when a causal connection is not included in both $F_{lower}{}^{f1}$ and $F_{lower}{}^{f2}$)

(4-2) Function blending in the FDP by verb and object conversion, when a causal connection is included in both $F_{lower}{}^{f1}$ and $F_{lower}{}^{f2}$ (see Figure 7)

$$
\begin{aligned}
&m_9 : F_{upper} \rightarrow F_{lower} \\
&f_1 \in BF_{lower}\,,\; f_1 = (S_1, V_1, O_1),\; f_2 = (S_2, V_2, O_2) \\
&| F_{lower}{}^{f2} | = n, \\
&f_i^{f1} \in F_{lower}{}^{f1},\; f_{n-1}^{f2} \in F_{lower}{}^{f2},\; f_n^{f2} \in F_{lower}{}^{f2},\; f_i^{f1} = (\overline{S_i}, \overline{V_i}, \overline{O_i}),\; f_{n-1}^{f2} = (\overline{S_{n-1}}, \overline{V_{n-1}}, \overline{O_{n-1}}),\; f_n^{f2} = (\overline{S_n}, \overline{V_n}, \overline{O_n}) \\
&when\; \overline{O_{n-1}} = \overline{S_n}, \overline{V_n} = V_2, \overline{O_n} = O_2, \overline{S_i} = \overline{S_{n-1}}, \overline{V_i} = \overline{V_{n-1}} \\
&f_{new} = (S_1, V_2, O_2), F_{lower} = \{ f_i \mid f_i \in (BF_{lower} - \{f_1\}) \cup \{f_{new}\} \}
\end{aligned} \tag{9}
$$

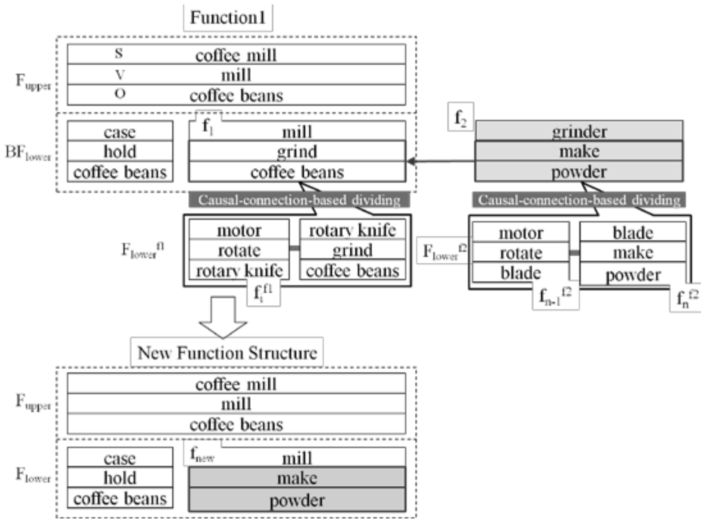

Figure 7. Example of function blending in the FDP by verb and object conversion, based on causal-connection-based dividing (when a causal connection is included in both $F_{lower}{}^{f1}$ and $F_{lower}{}^{f2}$)

3. CONSTRUCTING THESAURUS

Our proposed method is significant in that it facilitates identification of the hierarchal relations between the upper- and lower-level functions in the FDP (not involving the FBO), based on the linguistic hierarchal relations between the words on the upper-level function and the words on the lower-level functions [8]. In this method, only the noun and verb part-whole relations, which enable mapping between the upper- and lower-level functions, were investigated. As mentioned above, the superordinate-subordinate relation plays an important role in function blending. In addition, the made of/from relation (a relation between the engineering product and its material) is found in function blending as a result of the case study. Figure 8 shows the word relationships that compose the thesaurus supporting function blending in the FDP.

The inputs to the thesaurus are words (a noun or a verb, or both a noun and a verb) and a type of word relationship (a superordinate-subordinate relation or a part-whole relation). The outputs are pairs of words—the input word and another word, which is determined by the input relation to the input word. For example, when “product name” and “noun part-whole relation” are provided as input, the product name and its components are provided as an output pair.

The thesaurus is constructed in the following way. For the superordinate-subordinate relation, we use the relations that are contained in the existing dictionaries, since a sufficient number of relations are already contained by these dictionaries.

Regarding the part-whole relation, we need to develop a method for extracting the relations, since they are not sufficiently addressed by most of these dictionaries. Although we have proposed a method that extracts noun part-whole relations from patent abstracts in Japan by applying the *Espresso* algorithm [13], we need to consider a more detailed approach to extracting the verb part-whole relations.

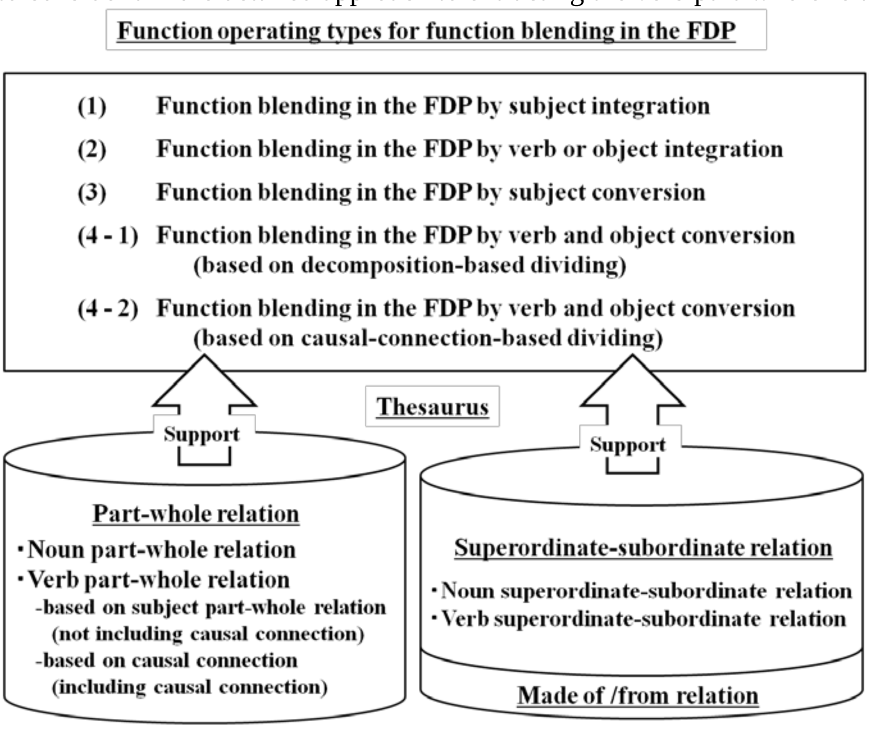

Figure 8. Relationship between function operating types for function blending in the FDP and the thesaurus

This is because the subject, verb, and object, which are elements of an expressed function, are not independent of each other. In particular, if a verb is stated independently, it is not meaningful. Accordingly, as with extracting part-whole relations for a verb, we have to consider the dependencies between the noun (subject and object) and the verb. Considering this dependency, we attempted to extract part-whole relations for a verb in the case where a causal connection between lower-level functions is not included [8]. In this attempt, we extracted the part-whole relations for a verb based on the part-whole relation of a subject, from documents. In other words, we collected the part-whole relations for a verb based on the part-whole relation of a subject by extracting the involved part-whole relations of a verb from the part-whole relations of nouns. Specifically, we extracted the verbs expressing a function from the collected documents whose headings include a part or whole of the noun, and we defined the relationship of the verbs. Table 2 lists examples of the obtained results. When the relation between a “vacuum cleaner” and a “fan” is a part-whole relation between the subjects, “suck up,” which is an action of a “vacuum cleaner,” and “blow,” which is an action of a “fan,” are determined to have a verb part-whole relation.

On the other hand, in the case where a causal connection between lower-level functions is included, it is necessary to extract the part-whole relations of a verb by considering not only the vertical relation between upper- and lower-level functions but also the horizontal relation between lower-level functions that have the causal connection. On the basis of the abovementioned previous study, in this study, we attempt to collect part-whole relations of verbs based on causal connection. The method for the extraction of the relations entails the following:

1. Extract the dependency relation between the subject and the verb from a set of documents.
2. Extract the dependency relation between the verb and the object from a set of documents.
3. Derive the part-whole relation of the verb, based on causal connection, by incorporating the dependency relations between the nouns and the verbs into the part-whole relation of the verb, which is determined from the part-whole relation of the subject.

The following are the procedures used to extract the dependency relation between a subject and a verb, and the dependency relation between a verb and an object.

1. Parse a set of documents.
2. Extract verbs that are found in the predefined functional verb patterns from the parsed documents.
3. Extract nouns that are part of a dependency relation, acting as subjects or objects.

The predefined functional verb patterns are expressed as follows:

to A, A *to B, which* A, *they* A, *that* A, *used to* A, *used for* A-*ing, by* A-*ing, device to* A.

Here, “A” and “B” represent verbs. These patterns can be found in manuals of engineering products. We used only “A” among these patterns. We extracted the dependency relations between subjects and verbs, and the dependency between verbs and objects. The verb B in the pattern “A to B” can be extracted by using the pattern “to A.”

The horizontal relation of verbs is extracted from causal connection on the basis of the matching of an object of a left-side function and a subject of a right-side function, between connecting lower-level functions, $f_1 = (S_1, V_1, O_1)$ and $f_2 = (S_2, V_2, O_2)$ (i.e., $O_1 = S_2$).

We extracted the dependency relations between subjects and verbs, and the dependency relations between verbs and objects, from patent abstracts in Japan and service manual texts (a total of approximately 8,000 lines). As a result, we extracted 11,198 dependency relations between subjects and verbs, and 13,348 dependency relations between verbs and objects. Table 3 lists examples of the extracted subject-verb dependency relations. Table 4 presents examples of the extracted verb-object dependency relations. Table 5 lists examples of the part-whole relation of verbs, based on causal connection. For example, for the upper-level function where the subject is “fan” and the verb is “produce,” lower-level functions (e.g., motor, rotate, impeller and impeller, produce, air) are causally connected, wherein a part-whole relation of verbs, based on a part-whole relation of subjects, is valid, between the upper-level function and each lower-level function.

4. SYSTEM FOR FUNCTION BLENDING IN THE FDP

A system for function blending in the FDP is composed of an input interface, an engine for the function blending in the FDP, an output interface, a thesaurus, and a case-based database. Figure 9 depicts an outline of the system for function blending in the FDP. We assume a usage scenario wherein a designer is combining two engineering products. The designer inputs two sets of subjects, verbs, and objects, in the form of (S, V, O), for the two engineering products. Two input functions are blended via the function dividing phase, word matching phase, and function blending phase, and finally, a new function structure is shown on the display. The thesaurus of part-whole relations supports the function dividing phase. The thesaurus of superordinate-subordinate relations and made-of/from relations supports the word matching phase and the function blending phase. The function structure obtained by the above process, using the system, is stored in the case-based database.

Table 2. *Examples of part-whole relation of verbs, based on a subject part-whole relation*

Subject part-whole relation (part : whole)	Verb part-whole relation (part : whole)
fan : vacuum cleaner	blow : suck up
motor : hair clipper	drive : cut
sensor : rice cooker	measure : cook
burner: cooking stove	burn : bake

Table 3. *Examples of dependency relation between subject and verb*

subject – verb
burner – burn, cam – rotate, filter – deodorize, heater – generate, rotor – drive, refrigerator – store, mirror – turn, holding member – hold, regulator – provide

Table 4. *Examples of dependency relation between verb and object*

verb – object
cut – hair, cool – engine, drive – blade, increase – pressure, pump – water, rotate – gear, rotate – impeller, turn – wheel

Table 5. *Examples of part-whole relation of verbs based on casual connection*

{upper-level function S, V : premise S, V, O upper-level function S, V : consequence S, V, O}
{ (fan, produce : motor, rotate, impeller), (fan, produce : impeller, produce, air) } { (transmission, change : control valve, actuate, clutch), (transmission, change : clutch, change, gear ratio) }

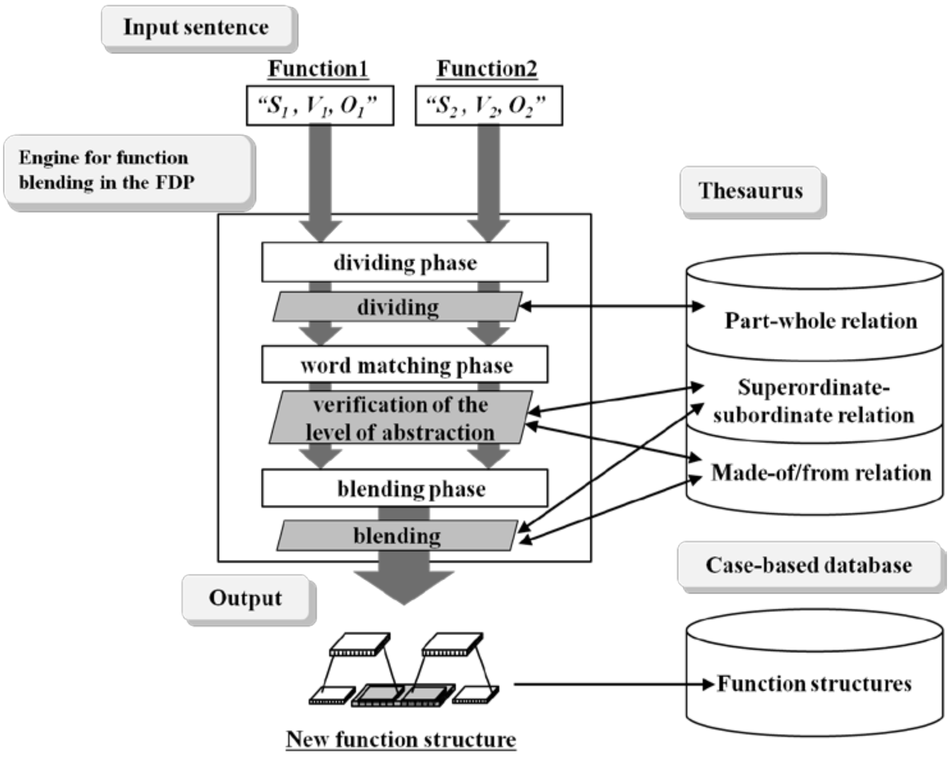

Figure 9. Outline of system for function blending in the FDP

5. DESIGN OF FUNCTION BY FUNCTION BLENDING IN THE FDP

To confirm the feasibility of the proposed methodology for function blending in the FDP, we conducted a trial to construct the function structure of a hair clipper with a vacuum cleaner, by blending a function of a vacuum cleaner and a function of a hair clipper. The thesaurus constructed by the authors was used. We tried to conduct the trial as naturally as possible, by not basing the operations on the thesaurus when the appropriate words were not found within it. Figure 10 shows the function structure constructed for the hair clipper with the vacuum cleaner. Verbs and nouns in italics represent those that exist in the thesaurus. In Figure 10, it can be seen that many relations between these italicized words are found in the thesaurus. This suggests that the thesaurus can support the process of function blending in the FDP. In particular, the method can support the design of a function. In the trial, by using “fan : vacuum cleaner – blow : suck up,” as in Table 2, a lower-level function; “fan blows dust,” was obtained from an upper-level function “vacuum cleaner sucks up dust.” In addition, by using “drive – blade” and “cut – hair,” as in Table 4, “hair clipper cuts hair” could be divided into “motor drives blade” and “blade cuts hair.” By using the superordinate-subordinate relations of verbs “rotate – move” and “drive – move,” along with the superordinate-subordinate relation of nouns “electric motor – motor,” a new function “motor moves impeller + blade” could be obtained by blending a lower-level function of the hair clipper, “motor drives blade,” and a lower-level function of the vacuum cleaner, “electric motor rotates impeller.” Although a hair clipper with a vacuum cleaner is an existing engineering product, we could derive a feasible function structure by using the proposed method of function blending in the FDP.

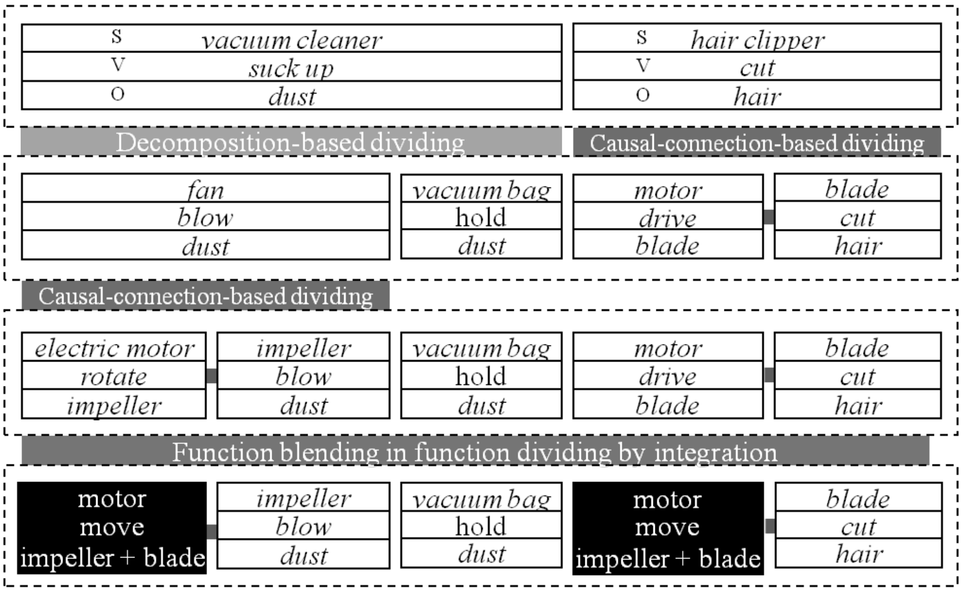

Figure 10. Constructed function structure of hair clipper with vacuum cleaner

6. CONCLUSION AND FUTURE WORK

In this study, we proposed a methodology for function blending in an FDP, which blends some functions to design a new function. First, we conducted a case study and derived a way to blend functions; that is, we identified operating types for function blending. Next, we formulated the identified operating types of the function blending in function dividing, from the viewpoint of linguistic hierarchal relations. Finally, we developed a thesaurus that is expected to support function blending in the FDP. The process of obtaining a function structure for an existing engineering product could be explained by using the procedures of function blending in the FDP. This suggests that our proposed method can support the design of a function. In the FBO, determining which two functions to blend is also an important issue. Regarding this issue, some of the authors have attempted to systematize a method for selecting functions to be blended; this has been reported in another paper [14]. In the future, with the aim of supporting the design of a new function, we will develop a more feasible system by considering these results.

REFERENCES

[1] Gero J.S. Design prototypes: A knowledge representation schema for design. *AI Magazine*, 1990, 11(4), 26-36.

[2] Taura T. and Yoshikawa H. A metric space for intelligent CAD, In *Intelligent Computer Aided Design - Proceedings of the IFIP WG5.2 Working Conference on Int CAD 91* -, 1992, Brown D.C., Waldron M.B., and Yoshikawa H. (eds.), North Holland, Amsterdam, NL, pp.133-157.

[3] Hirtz J., Stone R., Mcadams D., Szykman S., and Wood K. A functional basis for engineering design: Reconciling and evolving previous efforts. *Research in Engineering Design*, 2001, 13(2), 65-82.

[4] Nagel J.K.S., Stone R.B., and McAdams D.A. An engineering-to biology thesaurus for engineering design. *Proceedings of 2010 ASME IDETC/CIE*, Montreal, DETC2010-28233, 2010.

[5] Chiu U. and Shu L.H. Using language as related stimuli for concept generation. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 2007, 21(2), 103-121.
[6] Taura T., and Nagai Y. Discussion on direction of design creativity research (Part 1) - New definition of design and creativity: Beyond the problem-solving paradigm. In *Design Creativity 2010*, 2010, (Taura T. and Nagai Y. (eds.), Springer-Verlag), pp.3-8.
[7] Nagai Y., Taura T., and Mukai F. Concept blending and dissimilarity: factors for creative concept generation process. *Design Studies*, 2009, 30(6), 648-675.
[8] Yamamoto E., Taura T., Ohashi S., and Yamamoto M. A method for function dividing in conceptual design by focusing on linguistic hierarchal relations. *Journal of Computing and Information Science in Engineering*, 2010, 10(SEPTEMBER), 031004-1-8.
[9] Pahl G. and Beitz W. *Engineering Design-Systematic Approach*, 1995 (Springer-Verlag, Berlin).
[10] Miles L.D. *Techniques of Value Analysis and Engineering*, 3rd Edition, 1989 (McGraw-Hill, New York).
[11] Yoshikawa H. *General design theory and a CAD system*. In *Man-machine Communication in CAD/CAM*, 1981 (Sata E. and Warman E. (eds.), North Holland, Amsterdam), pp.35-58.
[12] Minami K., Taura T., and Tsumaya A. A study on modeling of reasoning process considering latent function. *Proceedings of Design and Systems Division Conference (JSME)*, Tokyo, October 2010, pp.202-207 (in Japanese).
[13] Pantel P. and Pennacchiotti M. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, 2006, pp.113-120.
[14] Sakaguchi S., Tsumaya A., Yamamoto E., and Taura T. A method for selecting base functions for function blending in order to design functions. *Proceedings of the 18th International Conference on Engineering Design* (*ICED11*), Copenhagen, 2011.

Contact: Toshiharu Taura
Kobe University
Organization of Advanced Science and Technology
1-1, Rokkodai, Nada 657-8501, Kobe, Japan
Phone/Fax: Int +81-78-803-6503
E-mail: taura@kobe-u.ac.jp

Toshiharu Taura is a vice-dean and professor in the Organization of Advanced Science and Technology (and professor in the Department of Mechanical Engineering) at Kobe University. He is currently working on several research themes that focus on the creative thought process of both engineering and industrial design, including the interdisciplinary aspects of design science.

Yu Park is a graduate student in the Department of Mechanical Engineering at Kobe University. He is studying conceptual design by focusing on functional operation.

Shota Ohashi conducted this study as a graduate student in the Department of Mechanical Engineering at Kobe University. Now, he is working for KDDI in Japan.

Eiko Yamamoto has conducted this study as an associate Professor (Lecturer) in the Department of Mechanical Engineering at Kobe University. Her interests include concept design supporting systems, impression analysis, information extraction, and knowledge acquisition based on natural language processing. She is currently working for Gifu Shotoku Gakuen University as an associate Professor.