View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Springer - Publisher Connector

Res Eng Design (2010) 21:99–111
DOI 10.1007/s00163-009-0080-7

ORIGINAL PAPER

On the conversion of functional models: bridging differences between functional taxonomies in the modeling of user actions

Dingmar van Eck

Received: 21 January 2009 / Revised: 29 September 2009 / Accepted: 1 October 2009 / Published online: 21 October 2009
© The Author(s) 2009. This article is published with open access at Springerlink.com

Abstract In this paper, I discuss a methodology for the conversion of functional models between functional taxonomies developed by Kitamura et al. (2007) and Ookubo et al. (2007). They apply their methodology to the conversion of functional models described in terms of the Functional Basis taxonomy into functional models described in terms of the Functional Concept Ontology taxonomy. I argue that this model conversion harbors two problems. One, a step in this model conversion that is aimed to handle differences in the modeling of user features consists of the removal of Functional Basis functions. It is shown that this removal can lead to considerable information loss. Two, some Functional Basis functions that I argue correspond to user functions, get re-interpreted as device functions in the model conversion. I present an alternative strategy that prevents information loss and information change in model conversions between the Functional Basis and Functional Concept Ontology taxonomies.

Keywords Engineering design · Functional modeling · Knowledge exchange · User actions

1 Introduction

As can be seen in a current review by Erden et al. (2008), engineering design research has produced an impressive wealth of functional modeling approaches. In these approaches also, a variety of definitions of functions, representations for functions, and strategies for decomposing functions into sub functions are proposed. For instance, Chakrabarti (1998) and Deng (2002) distinguish functions corresponding to intended behaviors from functions corresponding to purposes. With regard to the representation of functions, Chakrabarti and Blessing (1996) identify three frameworks that are in use in engineering: verb-noun representations, input–output flow transformations, and input–output state transformations. Exponents of these representational frameworks are, for instance, the function–behavior–state approach of Umeda et al. (1996) in which verb–noun representations are used, the systematic approach of Pahl and Beitz (1988) in which input–output flow transformations are employed, and the adaptive design approach of Goel and Stroelia (1996) in which functions are represented by input–output state transformations. More recently, Deng et al. (2000a, b) and Deng (2002) have added to this representational diversity by proposing the concepts of action and input–output flow of action transformation to represent functions. Concerning the decomposition of functions into sub functions, Van Eck et al. (2007) distinguish strategies in which functional decompositions are developed in a solution-neutral fashion from strategies in which known technical solutions for sub functions are incorporated from the outset. In addition, Goel et al. (2009) distinguish the modeling of artifacts qua "teleological systems" in which functions are realized by causal processes that mediate between function and structure, from the modeling of artifacts in which functions directly emerge from the shape of an artifacts' structure. One may conclude from the works mentioned earlier that many flowers bloom in the functional modeling segment of engineering design research.

D. van Eck (✉)
Department of Philosophy, Delft University of Technology, Jaffalaan 5, 2628 BX Delft, The Netherlands
e-mail: D.vanEck@tudelft.nl

A current research theme within functional modeling research concerns the development of methods that support the exchange and sharing of the functional knowledge, both between engineering design teams and between members of design teams. To support these tasks, different methods are proposed. For instance, Nagel et al. (2007a) have proposed a functional grammar to formalize functional modeling, and Szykman et al. (2001), Zhang et al. (2005), Kitamura and Mizoguchi (2004), and Kitamura et al. (2005/2006) have proposed function ontologies to archive and exchange functional knowledge. With regard to the exchange and sharing of functional knowledge, one can identify a reservation in the engineering literature though: some observe that the concept of function is ambiguous. Chandrasekaran and Josephson (2000), for instance, state that this ambiguity hampers the automation of function-based reasoning tasks, and Deng (2002) remarks that this ambiguity undermines the interchange of research ideas. A second reservation one can make is that the aforementioned methods for knowledge exchange are framed *within* the confines of a specific functional modeling approach or taxonomy, each with their own definitions of functions and schemes for representing functions. Although knowledge exchange is facilitated by shared representational schemes, if the observations of Chandrasekaran and Josephson (2000) and Deng (2002) carry weight, challenges may emerge when the aim is to establish knowledge exchange *between* different functional taxonomies or approaches. Kitamura et al. (2007) and Ookubo et al. (2007) have developed a methodology to support such knowledge exchange between different functional taxonomies. Their method to establish this is by converting functional models between functional taxonomies. Ookubo et al. (2007) apply this method to a conversion of functional models described in terms of the Functional Basis taxonomy of Stone and Wood (2000) into functional models described in terms of the Functional Concept Ontology taxonomy of Kitamura et al. (2005/2006).

In this paper, I review this model conversion and argue that ambiguities surrounding functional representations pose challenges for the conversion of functional models between the Functional Basis and Functional Concept Ontology taxonomies. More specifically, I argue that the model conversion leads to a number of problems. First, conceptual differences in the modeling of (parts of) users of devices, which are modeled in Functional Basis models and not in Functional Concept Ontology models, are handled by Ookubo et al. (2007) by removing Functional Basis functions that have input or output flows of human materials. I show that this removal can lead to considerable information loss in specific model conversions. Second, Hirtz et al. (2002) present the Functional Basis taxonomy as a taxonomy in which solely functions carried out by devices are described, and Ookubo et al. (2007) interpret the Functional Basis in a similar device-oriented fashion. I argue, however, that functions in the Functional Basis taxonomy may refer to functions of devices and to functions of user actions. I argue for this claim by analyzing Functional Basis models of a stapler, presented by Stone et al. (2004), and a power screwdriver, presented by Stone et al. (1998 and 2000). I show that the device-oriented perspective of Ookubo et al. (2007) on the Functional Basis taxonomy leads them to re-interpret Functional Basis functions corresponding to user actions as functions carried out by devices, and that this re-interpretation leads to information change in specific model conversions.

I then propose an alternative strategy to handle differences in the modeling of user aspects between the Functional Basis and the Functional Concept Ontology taxonomies, which addresses both these problems of information loss and information change. My main aim in this paper is to propose a strategy that supports improved knowledge exchange between these taxonomies. In a more speculative discussion, I suggest that this alternative strategy may also offer a solution for two additional research issues, currently investigated by Ookubo et al. (2007), in model conversions between the Functional Basis and Functional Concept Ontology taxonomies.

The paper has the following organization. The Functional Basis and the Functional Concept Ontology approaches are presented in Sect. 2. The model conversion methodology is presented in Sect. 3, where the methodology is illustrated with a discussion of a conversion of a functional model of a stapler. The removal solution is further analyzed in Sect. 4, and the user action analysis is presented there. These issues are illustrated with a discussion of conversions of functional models of a stapler and a power screwdriver. I then present my alternative strategy in Sect. 5. I suggest how the proposed strategy may solve other research issues in the model conversion in Sect. 6. I conclude the paper in Sect. 7.

2 Functional modeling approaches

2.1 The functional basis approach

The Functional Basis (FB) approach, formulated by Stone and Wood (2000), is an approach to functional modeling that is aimed at creating a common and consistent functional design language, dubbed a functional basis. This language allows designers to model overall product functions as sets of interconnected sub functions. The FB approach is focused on especially the electromechanical and mechanical domains. The approach is presented as supporting the archiving, comparison, and communication of functional descriptions of existing products, as well as the engineering designing of new products. Since the approach was proposed, it has been developed further. It is for instance used to develop a method to identify modules from functional models (Stone et al. 2000). It is also used to build a web-based repository in which functional decompositions of existing products are archived, as well as components counting as design solutions for the sub functions that are part of these decompositions.[1] The function and flow information of components archived in this repository has recently been employed by Bryant et al. (2007) in building a function-based component ontology. In this ontology product components are classified based on their most commonly ascribed sub functions as archived in the repository. The FB has also been extended by Nagel et al. (2007b) to domains outside engineering design proper, using the FB language to model manual processes.

In the FB approach, an overall product function refers to a general input/output relationship defined by the overall task of the product. This overall product function is described in a verb-object form and represented by a black-boxed operation on flows of materials, energies, and signals. A sub function, describing a part of the product's overall task, is also described in a verb-object form but represented by a well-defined basic operation on a well-defined basic flow of materials, energies, or signals. The black-boxed operations on general flows representing product functions are derived from customer needs, and the basic operations and basic flows representing sub functions are laid down in common and limited libraries that span the functional design space. These libraries are called a *functional basis*. In 2002, the FB approach was reconciled with an approach developed by Szykman et al. (1999) and coined Reconciled Functional Basis (Hirtz et al. 2002).

Stone and Wood (2000) present a three-step methodology to develop functional models or functional decompositions of products. The methodology starts with describing a product function in a verb-object form, represented by a black-boxed operation on flows of materials, energies, and signals. A chain of operations-on-flows is then specified, called a function chain, for each black box input flow, which transform that flow step-by-step into an output flow. These operations-on-flows are to be selected from the FB libraries. Finally, these temporally ordered function chains are aggregated into a single functional model of a product.

A FB model of a hand-held stapler is shown in Fig. 1, adapted from Stone et al. (2004). This model consists of temporally ordered chains of sub functions that transform the material input flows of "hand", "staples" and "sheet", and the energy input flow of "human force", step-by-step into output flows.

[1] http://function.basiceng.umr.edu/delabsite/repository.htmlref.

2.2 The functional concept ontology approach

The Functional Concept Ontology (FCO) approach, developed by Kitamura and Mizoguchi (2003, 2004) and Kitamura et al. (2005/2006), is an approach to functional modeling that is aimed at facilitating the sharing of engineering functional knowledge. In this approach, in order to facilitate knowledge exchange, a set of modeling guidelines and a functional modeling language are developed to assist the systematic and reusable description of functional models of devices. The approach supports various tasks. It is for instance employed in building an ontology for functions and in developing an automated design support system (Kitamura and Mizoguchi 2003). The approach is currently deployed in an engineering division of a Japanese industrial firm for sharing functional device knowledge among its team members (Kitamura et al. 2005/2006).

In the FCO approach, both behavioral models and functional models of devices are developed concurrently. Behaviors of devices and their components are defined as input–output relations between operand states. Operands refer to energy, fluid, material, motion, force, or information. Behaviors are represented as input–output state changes of properties of operands. Both overall functions and sub functions of devices are defined as roles played by behaviors, intended by designers or by users. Functions and sub functions are represented in terms of verb-operand pairs. The functional modeling language used in this approach consists of a generic set of verbs. These verbs are called functional concepts (Kitamura and Mizoguchi 2003; Kitamura et al. 2005/2006).

In a functional model or functional decomposition, a set of sub functions is specified that realize the overall function. Sub functions and overall functions are represented in terms of functional concepts. In a functional decomposition, it is furthermore specified by means of which technical principles the sub functions achieve the overall function. These specifications are referred to as "way of achievement" (Kitamura et al. 2005/2006).

A FCO model of a stapler is shown in Fig. 2, adopted from Ookubo et al. (2007). This model consists of the overall function of the stapler, and sub functions of the modules and components of the stapler. Ways of achievement are shown in the model, specifying how the component functions realize the module functions, and how the module functions realize the overall function. The module function "combine sheets and staples", for instance, contributes to the realization of the overall function "combine sheets" by an "intermediate way" that represents the combining of paper sheets via staples acting as intermediates between the sheets.

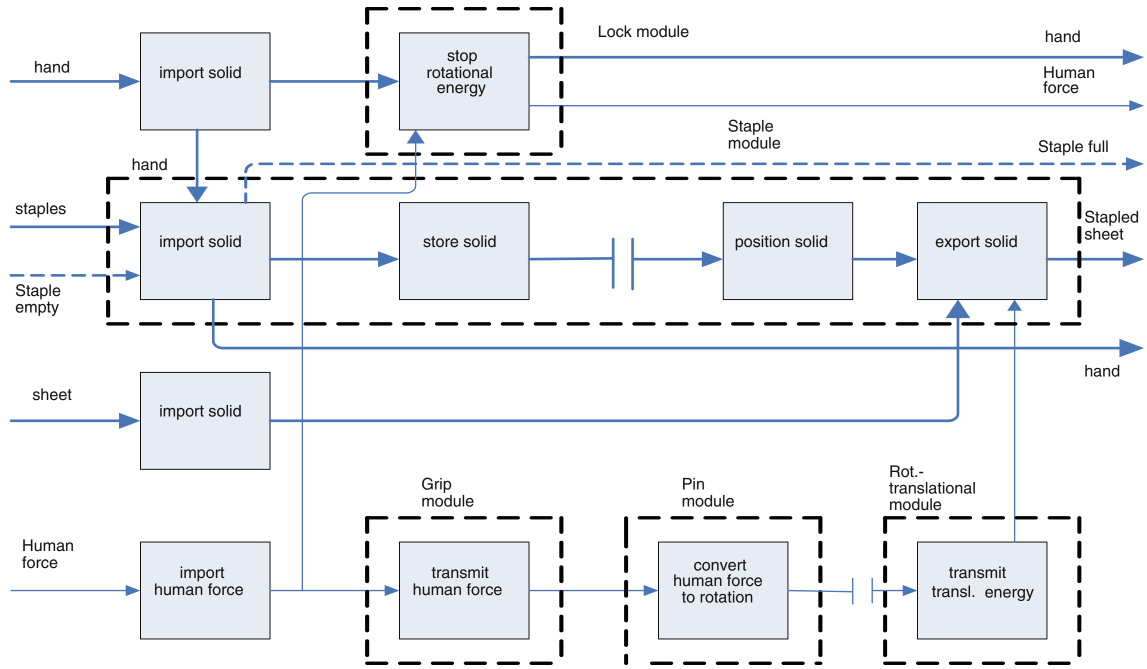

Fig. 1 FB model of stapler adapted from Stone et al. (2004)

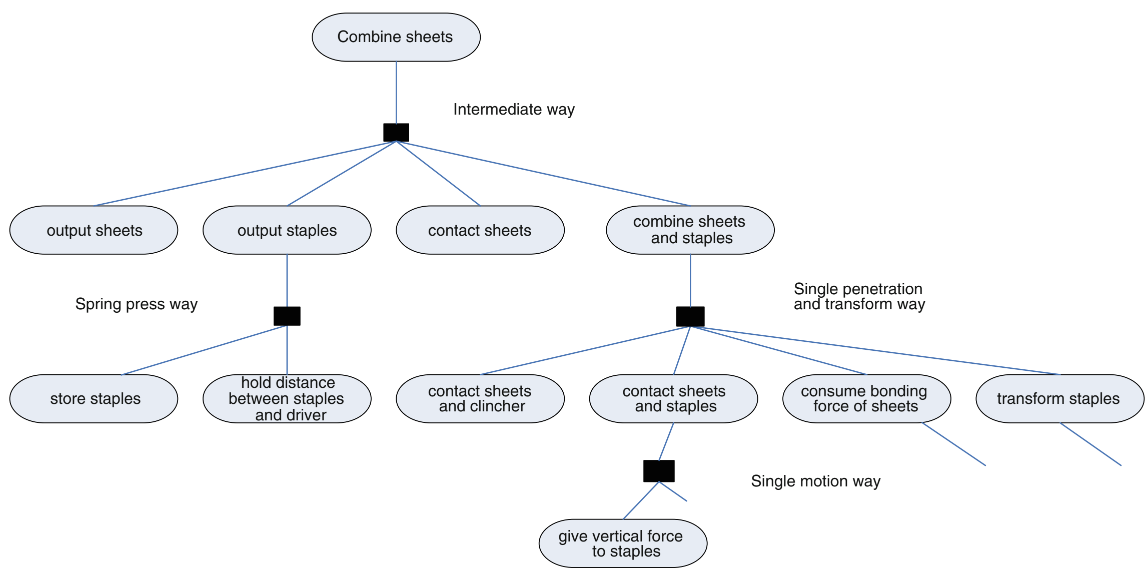

Fig. 2 FCO model of stapler adopted from Ookubo et al. (2007)

3 Functional model conversions

3.1 The conversion methodology

Kitamura et al. (2007) and Ookubo et al. (2007) aim with their methodology to support the conversion of a functional model *fm1*, which is based on one functional taxonomy *fx1*, into a (converted) functional model *fm2*, which is based on another functional taxonomy *fx2*. Functional models are converted by carrying out two steps. In the first step, the function terms of *fx1* are translated into function terms of the other taxonomy *fx2*. By this *fx1*-to-*fx2* function term translation, function terms in *fm1* are translated into function terms that will be included in *fm2*. In the second step, conceptual differences between models based on *fx1* and models based on *fx2* are explicated, and measures are developed and carried out to minimize these in the model conversion. By minimizing these differences, Ookubo et al. (2007) and Kitamura et al. (2007) aim to improve knowledge exchange between *fx1* and *fx2*. After these translation and modification steps, a functional model *fm1* based on *fx1* is converted into a functional model *fm2* based on *fx2*.

In the first step, function terms are translated by using a "reference ontology" for functions (Kitamura et al. 2007; Ookubo et al. 2007). This reference ontology is used to identify the meaning of functions that are part of functional taxonomies and, based on this identification, to translate functions between taxonomies. In this reference ontology, function categories are defined which are stated to correspond to existing engineering meanings of functions. Definitions of these function categories are based upon the conceptual structure of the FCO approach (Kitamura et al. 2007; Ookubo et al. 2007). The FCO concepts of device, behavior, function, and operand are further specified into subtypes called "descriptors of functions" (Kitamura et al. 2007). With these descriptors of functions, different function categories are defined in the reference ontology. With these function categories they aim to identify different meanings of the concept of function in the engineering domain.

According to Kitamura et al. (2007) and Ookubo et al. (2007), by first classifying the function terms from *fx1* and *fx2* into function categories, their meaning can be established. This classification is done by matching the definitions of function terms of *fx1* and *fx2*, as laid down in *fx1* and *fx2*, with the function categories in the reference ontology. The function terms in *fm1* are then translated into function terms that will be part of *fm2*. Depending on how these function terms are classified, different sorts of translations are carried out. Translations between function terms that are classified in the same function category are presented as straightforward, since the same meaning is attached to these function terms. These translations are called "within category" mappings. When *fx1* includes function terms that are classified in a certain function category and *fx2* lacks function terms that can be classified in that same function category, translating these function terms from *fx1* to *fx2* involve more complex procedures. Such function terms (and their meaning) are namely part of one taxonomy, but not part of the other taxonomy (Kitamura et al. 2007; Ookubo et al. 2007). These more complex translations are called "between category" mappings.

After this first translation step, an interim functional model *fm** results consisting of translated function terms that are represented in terms of *fx2*. In this phase, *fm** still has the same model structure as *fm1*, i.e., all the model features of *fm1* are also represented in *fm**. In the second step, conceptual differences between models based on *fx1* and models based on *fx2* are further explicated. This is done by comparing *fm1* with a functional model of the same device that is described in terms of *fx2* functions and according to *fx2* modeling criteria. Let us abbreviate this comparison model as *fmC*. The conceptual differences identified between *fm1* and the comparison model *fmC*, are then used to modify *fm**, resulting in *fm2*. After these translation and modification steps, a functional model *fm1* based on *fx1* is converted into a functional model *fm2* based on *fx2*.

3.2 The methodology at work: an FB-to-FCO model conversion

Ookubo et al. (2007) demonstrate their method by a conversion of an FB model (*fm1*) of a stapler represented in terms of the FB taxonomy (*fx1*) into a model (*fm2*) represented in terms of the FCO taxonomy (*fx2*). They also use a comparison FCO model of a stapler (*fmC*) in this conversion. This comparison FCO model (*fmC*) is used to identify conceptual differences between models based on the FB taxonomy and models based on the FCO taxonomy. The FB model (*fm1*), which Ookubo et al. (2007) adapted from Stone et al. (2004), is shown in Fig. 1, the comparison FCO model (*fmC*) is shown in Fig. 2, and the converted FCO model (*fm2*) is shown in Fig. 3.[2]

3.2.1 Step 1: translating FB function terms into FCO function terms

In the first step of the model conversion, Ookubo et al. (2007) translate functions both by "within category" mappings and by "between category" mappings. Most FB function terms and all FCO function terms are classified in the "flowing object" function category (Kitamura et al. 2007). Flowing object functions correspond to a specific type of behavior, to wit: temporal changes in attributes of a physical entity, such as matter and energy flows or operands, within a device's system boundary. A role is attached to these behaviors in a teleological context (Kitamura et al. 2007; Ookubo et al. 2007). Since most function terms in the FB and FCO taxonomies are classified as flowing object functions, the same meaning is attached to them.

[2] I present the same adaptation of the FB model as Ookubo et al. (2007) give. This adaptation consists in excluding several operations-on-flows which are described in the original FB model. The vertical lines intersecting the "human force" flow and the "staples" flow represent this exclusion. In addition, to be precise, the converted FCO model is a converted FB model, expressed in terms of the FCO taxonomy. Please note that I use the term "converted FCO model" for brevity, but that this term has the meaning expressed above. Finally, Ookubo et al. (2007) use the concept of an "interim functional model" (Sect. 3.1) at a conceptual level, but do not give an example of such a model. I follow their usage of this concept here.

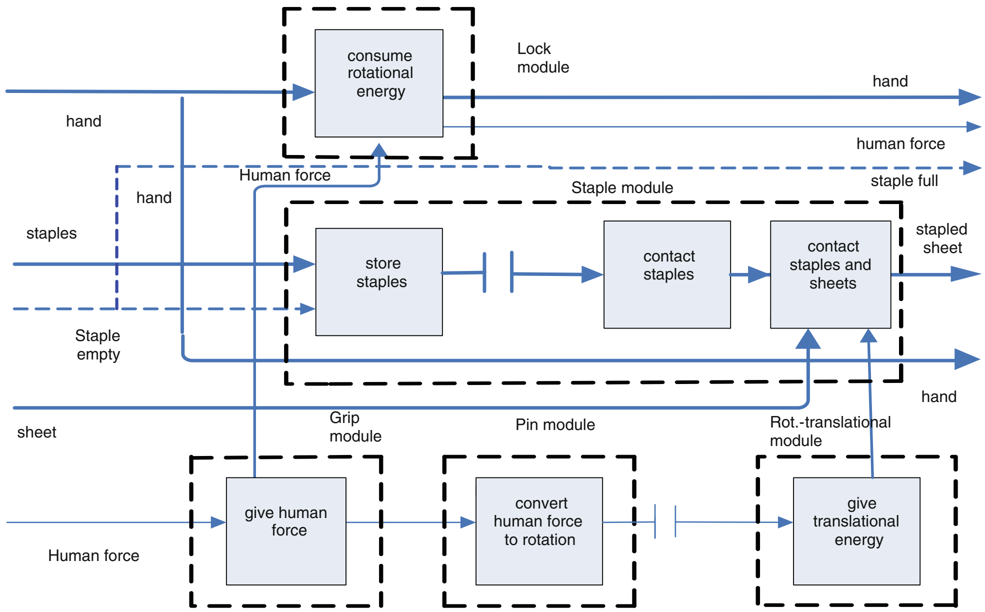

Fig. 3 Converted FCO model of stapler adopted from Ookubo et al. (2007)

These function terms are translated by "within category" mappings. An example of a within category mapping of flowing object functions in the model conversion is the FB function "transmit human force" (Fig. 1) that is translated into the FCO function "give human force" (Fig. 3).

Some of the FB function terms in the FB stapler model are classified in the reference ontology as "system interface functions". System interface functions represent temporal changes in attributes of a physical entity on a system boundary. The FB "import" and "export" function terms are classified as system-interface functions. Since the FCO solely consists of "flowing object functions" (Ookubo et al. 2007), these FB function terms are translated by a between category mapping: the FB "import" and "export" operations-on-flows are translated in the model conversion into FCO input and output operands. Examples of between category translations are the "import solid (sheet)" and "export solid (stapled sheet)" functions of the FB model (Fig. 1) that Ookubo et al. (2007) represent in the converted FCO model as input and output operands of "sheet" and "stapled sheet" (see Fig. 3). This first translation step establishes a model (*fm**) in which translated functions are described in terms of the FCO taxonomy, but the model still has the structure of the FB model.

Other function categories into which FB function terms are classified are the "function with way of achievement" function category and the "composite device" function category. Function terms of the FB model of the stapler are not classified in these categories, but I describe them here for sake of completeness. FB function terms classified as functions "with way of achievement" correspond to a flowing object function but in addition also refer to a way of achievement. An example given by Ookubo et al. (2007) is the FB term "link", which has both the (flowing object function) meaning of "coupling flows together" and also refers to how this coupling is achieved, namely by an "intermediary flow". FB function terms classified as "composite device" functions correspond to a flowing object function and the meaning of the term, as defined in the FB taxonomy, can be interpreted in two different ways viewed from the FCO taxonomy. An example given by Ookubo et al. (2007) is the FB term "guide" which they interpret as either referring to "supply motion" or to "change direction of motion".

3.2.2 *Step 2: modifying the interim model*

After these translations, the FB model (*fm1*, Fig. 1) and the comparison FCO model (*fmC*, Fig. 2) are compared in the second step to identify conceptual differences between these models. Based on these differences, procedures are then developed to modify the interim model (*fm**). The end result of these translations and modifications is the converted FCO model (*fm2*, Fig. 3). Six conceptual differences are identified between the FB model and the comparison FCO model (Ookubo et al. 2007):

1. In FCO models, overall functions are related to sub functions of modules, which are related to sub functions of components. FB models do not represent relationships between sub functions of modules and components.

2. In FCO models, functions are not connected by operands, whereas functions are connected by flows they have as input or output in FB models.
3. In FCO models, ways of achievements are described, whereas these are not described in FB models.
4. In FCO models, changes in distance between physical objects—matter and energy flows/operands—are described, whereas these are not described in FB models.
5. In FCO models, features of users are not described, whereas features of users are described by human material flows in FB models.
6. In FCO models, material and energy operands may be grouped together in descriptions of functions, whereas material and energy flows are separated in FB models.

In the conversion of the FB stapler model, Ookubo et al. (2007) develop and carry out modifications to handle the difference in distance changes between physical objects (4) and to handle the difference in user features (5). They are currently investigating modifications to handle the difference in connections between functions (2) and to handle the difference in separating versus grouping material and energy (6). The converted FCO model in Fig. 3 thus is the result from the translation of functions in the first step and from the modifications in the second step that address differences in representing distance changes between flows/operands, and differences in representing (parts of) users of devices. This model is currently the endpoint of the conversion process (Ookubo et al. 2007).

The difference between the FB model and the comparison FCO model concerning distance changes between flows/operands is handled by adding an FCO function from the comparison model to the interim model. In the FB model, the "staple" and "sheet" flows enter the stapler as separate flows and exit as the combined flow "stapled sheet". The combining of these flows, referring to a change in distance between flows, is not represented in the FB model. In contrast, this combining is explicitly represented in the comparison FCO model by the function "contact staples and sheets". This difference is handled by adding this FCO function of the comparison model to the interim model.

The difference between the models regarding the representation of (parts of) users of devices is handled by *removing* FB functions in the model conversion. In the FB model, parts of users are represented in terms of flows of human materials such as "hand". In contrast, parts of users are not represented in the comparison FCO model, nor are they in FCO models of devices in general. The FCO treats (parts of) users as external to devices and therefore does not represent these in functional models of devices. Ookubo et al. (2007) handle this difference by removing FB functions that have input or output flows of human materials. In the interim model, for instance, the FB function "import solid (hand)" is removed.[3]

The end result of these translations and modifications is the converted FCO model (*fm2*) in Fig. 3. As can be seen, the FCO function "contact staples and sheets" is added to this model, and the FB function "import solid (hand)" is removed from this model.

4 Problems concerning user functions

In this section, I argue that the FB-to-FCO model conversion, interesting though it is, leads to a number of problems. One, the removal of FB functions that have input or output flows of human materials may lead to considerable information loss. I argue that the converted FCO model of the stapler is a case in which the loss of information is limited. I then present an example of a FB model of a power screwdriver that gives a more extreme illustration of this information loss. Two, not all FB functions involving human material flows are actually removed in the stapler model conversion. I argue that such a partial application of the removal solution may lead to function-to-function translations in which the meaning of some FB functions is altered. Ookubo et al. (2007) interpret the FB as modeling only functions performed by devices. However, I will argue that in the FB model of the power screwdriver some of the operations on human material flows represent user actions. If the device-oriented perspective of Ookubo et al. (2007) on the FB is maintained in the screwdriver case, re-interpretations of FB functions that correspond to user actions as functions carried out by devices will occur. This results in information change.

These problems lead to either information loss or information change, limiting the establishment of knowledge exchange and interoperability between taxonomies. I discuss these problems further in the sections below. I then present my alternative strategy in Sect. 5 and show that it prevents these problems.

4.1 Removing FB functions

In the stapler model conversion, the removal solution is only partly applied. First, a "hand" flow/operand is represented in the converted FCO model (*fm2*, Fig. 3). Second, the function "consume rotational energy" that transforms a "hand" flow/operand is still represented in the converted FCO model. When the removal solution would

[3] Ookubo et al. (2007) state that "the function whose input or output is part of the user as flow is removed as a result of the transformation" (p. 10).

have been strictly applied, the functionality of the lock module, represented by the "consume rotational energy" function, would be lost in the conversion as well, in addition to the loss of the "import solid (hand)" function. This would have resulted in (limited) information loss. The FB model of the screwdriver, shown in Fig. 4, gives a more extreme illustration of this information loss.

The FB functional model of the power screwdriver, described by Stone et al. (1998 and 2000), is shown in Fig. 4. Stone et al. (1998) state that the first function chain represents the insertion and removal of the screw bit, that the second represents the fastening of the screw bit, that the third represents the positioning of the screwdriver, and that the fourth represents the actuation of the device. The first and second function chains consist solely of functions that transform a (branching) human material "hand" flow from input to output. The third function chain consists of two FB functions that transform a "hand" flow into output.

If this screwdriver model would be selected for a model conversion, strict application of the removal solution will lead to the complete removal of the first three function chains. Consequently, a converted FCO model of the screwdriver will not represent the functionality of interchangeable screw bits, nor the functionality of the fastening/locking mechanism of the screw bit and neither the functionality of the positioning of the screwdriver.

Besides this information loss by removal of FB functions, a second problem may emerge in FB-to-FCO model conversions. In case of the screwdriver model, the misclassification of FB functions leads to information change.

4.2 Misidentifications of FB functions

Both the FB approach and the FCO approach are presented as device-oriented taxonomies. Ookubo et al. (2007) write that the FCO adopts "a device-oriented viewpoint" (p. 4) toward the modeling of functions of devices and components. Hirtz et al. (2002) also present the FB as device-oriented, by remarking that:

> "We judge a function term's suitability based on whether or not it describes an operation that a product or device carries out on a flow. This ensures that the reconciled functional basis will consist of only *device functions*, as opposed to *user functions*" (p. 72, italics in original)

Hirtz et al. (2002) illustrate the difference between device functions and user functions with an example of a coffee machine: a coffee machine imports a flow of water, while a user pours water into the device. In this example, they characterize the notion of a user function as an operation (pouring) carried out by a user on a flow (water). In other words, their characterization of a user function corresponds to a user action. The position taken by Hirtz et al. (2002) on the FB as modeling only device functions, and not user functions, is also adopted by Ookubo et al. (2007). They are

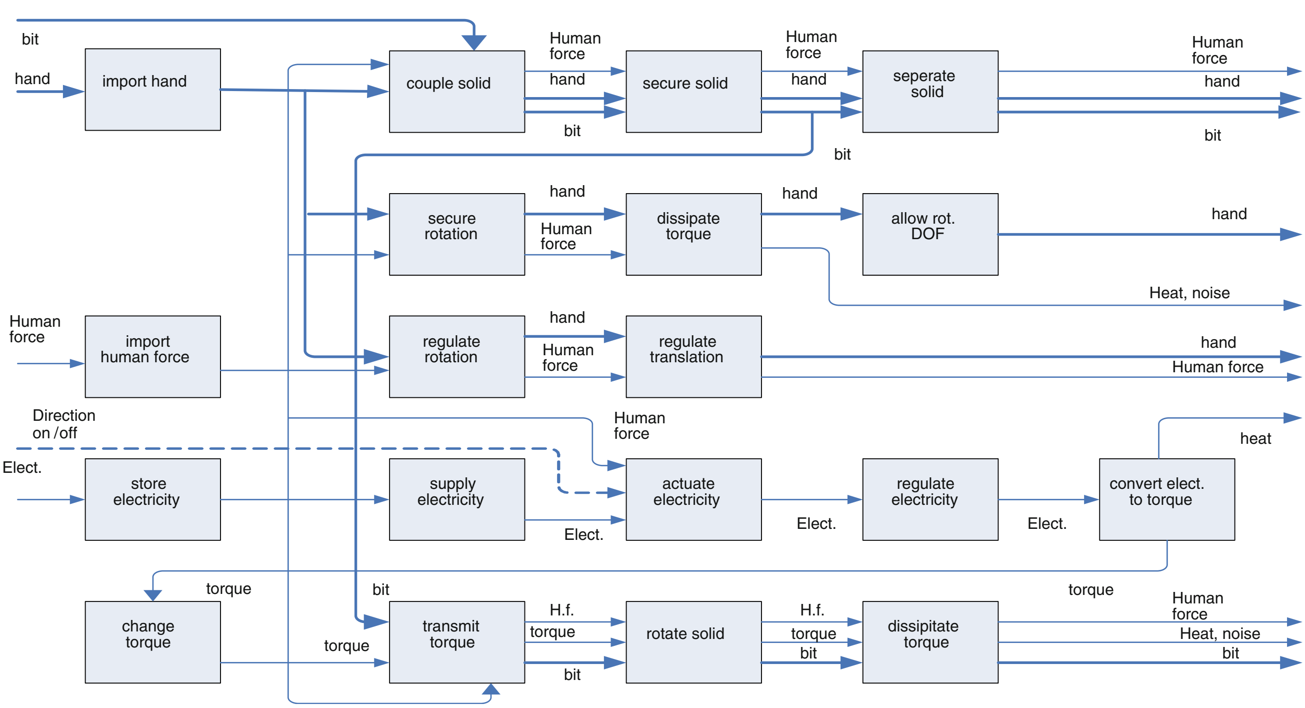

Fig. 4 FB model of screwdriver adopted from Stone et al. (1998 and 2000)

explicit in their device-oriented perspective on the FB approach: "our functional concept ontology shares a device-oriented viewpoint with FB" (p. 5). This perspective informs their classification of FB functions into function categories, which are all categories of functions implemented by devices (cf. Kitamura et al. 2007; Ookubo et al. 2007).

In my view, examples can be given that contradict the device-oriented perspectives of Hirtz et al. (2002) and Ookubo et al. (2007). For instance, the FB function "import solid (hand)", which is removed in the stapler model conversion, is a function performed by a user. Whereas Ookubo et al. (2007) classify this function as a "system interface function", the importing of the hand represents an operation carried out by a user on a flow. Returning to the FB model of the power screwdriver, however, much more functions that have input or output flows of human materials correspond to user functions.

I argue that all the FB functions of the first function chain and the leftmost function of the second function chain of the power screwdriver exemplify the characterization of user functions given by Hirtz et al. (2002). As can be seen in Fig. 4, the first function chain is represented in terms of four FB functions that transform the flows "hand", "bit", and "human force" from input to output. By representing the insertion and removal of the screw bit in terms of a sequence of FB functions that transform a material "bit" flow, a "human force" flow, and a "hand" flow, the (de)coupling of the screw bit is represented as *a sequence of user functions*. More specifically, the (de)coupling of the screw bit is realized through human force applied through the hand, i.e., operations-on-flows carried out by a user. This analysis applies as well to the leftmost function "secure rotation" of the second function chain, which represents the manual fastening of the screw bit. In this function chain, the FB function "secure rotation" transforms a "human force" flow and a "hand" flow, describing that the securing operation is realized by human force applied through the hand.

In this example, Ookubo et al.'s (2007) device-oriented perspective on the FB, given that they do not or partially apply their removal solution, results in information change. The above FB functions, identified as user functions, will be misclassified as functions carried out by devices. The device-oriented perspective put forward by Hirtz et al. (2002) and adopted by Ookubo et al. (2007) unfortunately leads to function-to-function translations in which the meaning of functions is altered.

I do not want to end on these critical notes however. Both the model conversion methodology and the FB and FCO approaches are too valuable and interesting to end with these critical observations. In the remaining part of this paper, I present a possible solution for the problems outlined above and apply it to both the converted FCO model of the stapler (Fig. 3) and to the FB model of the screwdriver (Fig. 4).

5 A strategy for translating user functions

In order to avoid information loss and information change, one can imagine the following alternative. I propose that after a model conversion, in which the removal solution is not applied, one can derive functions of devices and user functions from their corresponding operations-on-flows of the converted FCO model (*fm2*). These derived functions can then be represented in another FCO functional model. Whereas the converted FCO model (Fig. 3) is currently the endpoint of the conversion process in the proposal of Ookubo et al. (2007), I use it in my derivation strategy as an interim step for developing a derived FCO functional model. In this derived FCO model, both functions of devices and user functions are represented. The derived functions corresponding to functions of devices can be represented in terms of the FCO language. The derived functions corresponding to user actions can be represented in terms of an application of the function behavior representation language (FBRL), on which the FCO taxonomy is based, developed by Van der Vegte et al. (2004). Van der Vegte et al. (2004) apply the FBRL toward the modeling of user actions. If one accepts that FB functions may correspond to user actions and, hence, that translations of functions in FB-to-FCO model conversions may concern translations of user functions, this application becomes available as a means to represent user functions in a derived FCO model.

In the application of Van der Vegte et al. (2004), FBRL function verbs and operands are used to describe both functions carried out by devices and user actions with devices. Whereas in the former case a device is (somewhat confusingly) considered the "agent" of the function, in the latter the user is considered the "agent" of the function. For instance, in the case of a coffee machine, they describe functions of devices such as "conduct hot water" of a tube, and user actions such as "move basket" and "deform filter". In this extension of FBRL, models of user actions are represented separately from models of device functions. In contrast, my analysis of the FB screwdriver model shows that user functions are represented within this FB functional model.

Applied to the converted FCO model of the stapler, the result of my strategy is shown in Fig. 5. The derived FCO model in Fig. 5 has a similar format as the FCO comparison model (cf. Fig. 2), except that ways of achievement are not represented. As mentioned in Sect. 4.1, if the removal solution had been applied consistently by Ookubo et al. (2007), the functionality of the lock module would be

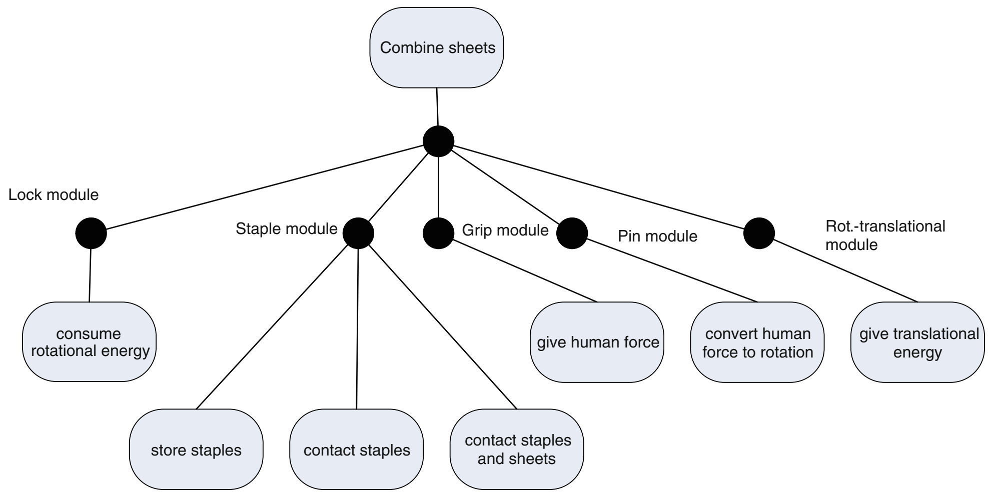

Fig. 5 Derived FCO model of stapler. The overall function "combine sheets" is the same as the FCO model of Fig. 2. The functions are represented according to their grouping in function chains in the converted FCO model (cf. Fig. 3). Module information is given at the nodes (cf. Fig. 3)

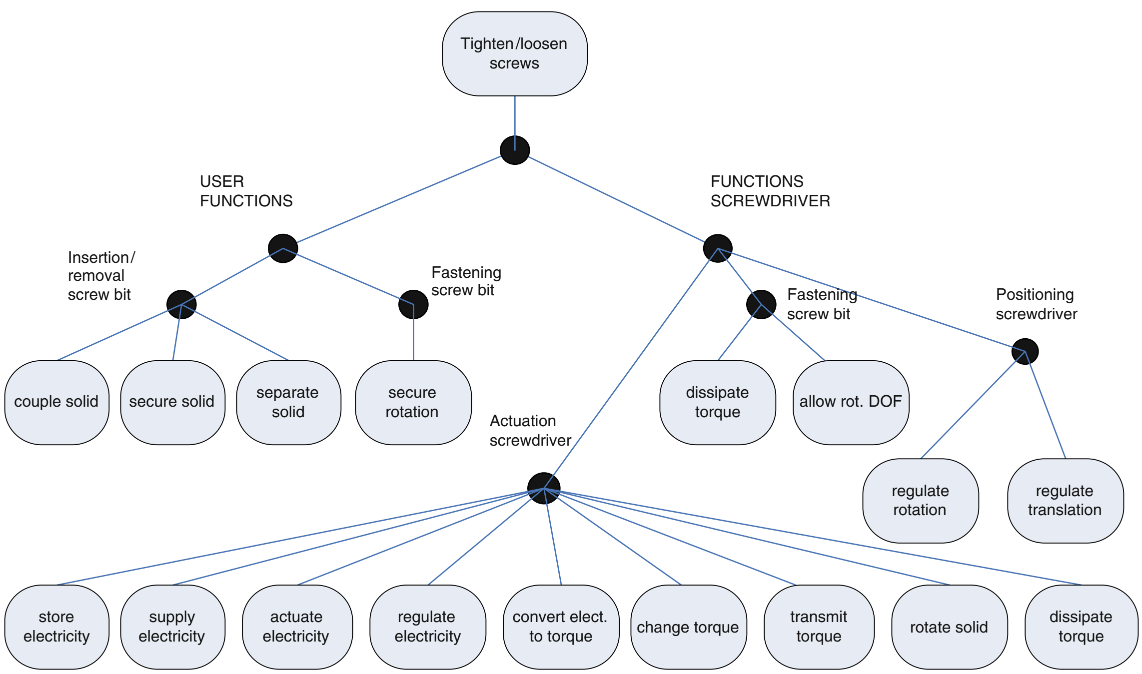

Fig. 6 Derived FCO model of screwdriver. The goal function "tighten/loosen screws" is the same as the product function of the screwdriver given by Stone et al. (1998 and 2000). User functions are separated from functions carried out by the screwdriver. The derived functions are represented according to their grouping in function chains in the FB model of the power screwdriver (cf. Fig. 4). The functionality of the function chains, adopted from Stone et al. (1998), is described at the nodes

lost in the conversion. With my strategy this functional information is preserved straightforwardly.

Applied to the FB screwdriver model, the result of my strategy is shown in Fig. 6. For brevity, I have omitted the step of presenting a converted FCO model. In the derived FCO model in Fig. 6 both the functions carried out by the screwdriver and the user functions with the screwdriver are described. The derived FCO model in my proposal has a similar format as the FCO comparison model (cf. Fig. 2), except that ways of achievement are not represented. In line with the FBRL user action application, the user functions are grouped together and separated from the functions of the device. This model thereby accords with the device-oriented perspective of FCO. In accordance with the FCO taxonomy and FB-to-FCO model conversions, functions corresponding to the "import" operations-on-flows "import hand" and "import human force" are not derived. In my derivation strategy, the flows that are present in a converted FCO model, including the human material flows/operands, are not represented in a derived FCO model. In my view, differences in the modeling of user aspects are with this strategy addressed, in a way that is congruent with the modeling guidelines of FCO, which does not entail information loss by the removal of FB functions (see Figs. 5, 6) and that does not change the meaning of FB user functions (see Fig. 6).

In line with the aim underlying the conversion methodology to establish functional knowledge exchange between taxonomies, I present my alternative as a conceptual tool to address the loss of and changes in functional information in FB-to-FCO model conversions. A general suggestion that may be drawn from the presented analysis is that the inclusion of a function category of user functions in the reference ontology would enhance the translation possibilities with the conversion methodology. Given that other functional modeling approaches are developed in which user functions are described, such as Otto and Wood's Reverse Engineering and Redesign approach (1998, 2001), this seems an extension worth considering. Inclusion of a user function category would enable the identification and translation of user functions between taxonomies. How to proceed in specific cases will depend on the specific taxonomies paired in a model conversion. The strategy described above gives a handle on this issue in the case of FB-to-FCO model conversions.

The solution that I have presented is a conceptual one and not empirically clear-cut. The method that I adopt in this paper is analytic, by which I mean the analysis of concepts and assumptions that are used in the functional modeling approaches and the conversion methodology. The advantage of this method, in my view, is that it is well suited for elucidating concepts. It is, however, less suited for empirical testing. I acknowledge this limitation and therefore leave validation of my proposal by the empirical experts.

The position that I developed above may have additional practical utility. It may offer a solution for two research issues in FB-to-FCO model conversions, currently investigated by Ookubo et al. (2007). And, in addition, my solution for these research issues may be extended to model conversions between other functional modeling approaches. I outline this solution in the next section.

6 Discussion: generalizing the derivation strategy

In this paper, I have focused my derivation strategy on the translation of user functions between the FB and FCO approaches, addressing functional information loss and information change. My strategy is not limited to these two approaches. It generalizes, for instance, straightforwardly to conversions between Otto and Wood's Reverse Engineering and Redesign approach (1998, 2001), in which both device functions and user functions are represented by operations-on-flows, and the FCO approach. Since in Otto and Wood's approach user functions are represented exactly the same as in the FB approach, these would be removed in the conversion methodology of Kitamura et al. (2007) and Ookubo et al. (2007). In contrast, my strategy enables their representation in a derived FCO model, thus preventing information loss.

The derivation strategy that I have described and demonstrated seems, in addition, an adequate tool for solving two research issues currently investigated by Ookubo et al. (2007), mentioned in Sect. 3.2. These research issues concern, first, the modeling of connections between functions in terms of flows that are modeled in FB models, but not in FCO models, and, second, the separation of material and energy in FB models, which, instead, may be combined in FCO models. For instance, the FCO description of the stapler's sub function "give vertical force to staples", in which energy and material are combined (Ookubo et al. 2007; cf. Fig. 2). With my strategy both these differences can be handled. Regarding the first issue, my proposed step of deriving device functions and user functions from their corresponding operations-on-flows in the converted FCO model leaves flow connections between functions in a converted FCO model. Thereby, flow connectivity information is preserved in an (interim) FB-FCO model conversion. Yet, in accordance with FCO modeling rules, this connectivity is not represented in a derived FCO model. My strategy thus handles this difference, without information loss (cf. Figs. 5, 6). Regarding the second issue, a solution can be developed along similar lines. The separation of material and energy in FB descriptions of functions follows from the fact that they are connected by separate material and energy flows, which they have as input and output. Since in a derived FCO model in my strategy, functions are not connected by material and energy flows, it is also no longer required that functions are described in terms of the separation of material and energy. In my strategy, one can thus take the separation of material and energy as a feature of converted FCO models, but not of derived FCO models. Thereby, information on material and energy separation is preserved in an FB-FCO model conversion. Yet, in accordance with FCO modeling rules, this separation is not represented in a derived FCO model. My strategy thus handles this difference, without information loss.

Taking these next steps in the above fashion in FB-FCO model conversions seems a promising way to tackle differences between other approaches in model conversions as well. The connectivity between functions and the separation of material and energy are two features that are highly discriminative between functional modeling approaches (see for instance the review by Erden et al. 2008). The specific details of such model conversion-cases will, of course, depend on the approaches paired in a model conversion. The strategy proposed here provides a general conceptual framework for developing them.

The examples of the stapler and screwdriver models discussed in this paper also highlight a general research challenge that must be addressed in model conversions between specific functional modeling approaches: when certain types of functions are modeled in one approach, but not in the other, measures need to be developed that prevent information loss and/or information change problems. For instance, in the Multi Level Flow approach of Lind (1994) functions may correspond to operator actions, whereas in the FB and FCO approaches they do not. This difference needs to be addressed in order to avoid information loss.

Future work is aimed at investigating in detail the issues raised in this discussion section. The main contribution presented in this paper concerns an alternative way of handling differences in the modeling of user aspects in FB-to-FCO model conversions that prevents the loss of and change in functional information.

7 Conclusions

In this paper, I have reviewed a methodology for the conversion of functional models between functional taxonomies developed by Kitamura et al. (2007) and Ookubo et al. (2007). They apply their methodology to the conversion of functional models described in terms of the Functional Basis taxonomy into functional models described in terms of the Functional Concept Ontology taxonomy. I have argued that these model conversions harbor two problems. One, a step in these model conversions that is aimed to handle differences in the modeling of user features is shown to lead to considerable information loss. Two, it is shown that Functional Basis functions that correspond to user actions get re-interpreted as functions carried out by devices, leading to information change. After this analysis, I have presented and demonstrated an alternative strategy for solving this information loss and information change. I ended the paper by outlining how my alternative strategy may also solve other research issues, both in model conversions between the Functional Basis and Functional Concept Ontology taxonomies and between other functional modeling approaches. Future work is aimed at testing the strategy in detail with respect to these research issues. At a more general level, the research presented here is submitted as a contribution toward the clarification of the meaning and representation of functions in engineering and toward the support of functional knowledge exchange between functional modeling approaches.

Acknowledgments I would like to thank the reviewers for their insightful comments. I would also like to thank Maarten Franssen, Yoshinobu Kitamura, Peter Kroes, Riichiro Mizoguchi, and in particular Pieter Vermaas for their incisive comments on earlier versions of this paper. This research is supported by the Netherlands Organization of Scientific Research (NWO).

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

Bryant CR, Stone RB, Greer JL, McAdams DA, Kurtoglu T, Campbell MI (2007) A function-based component ontology for systems design. In: Proceedings of international conference on engineering design (ICED 07), Paris, 28–31 August 2007

Chakrabarti A (1998) Supporting two views of function in mechanical designs. In: Proceedings 15th national conference on artificial intelligence, AAAI'98, Madison, 26–30 July 1998

Chakrabarti A, Blessing L (1996) Special issue: representing functionality in design. AIEDAM 10(5):251–253

Chandrasekaran B, Josephson JR (2000) Function in device representation. Eng Comput 16:162–177

Deng YM (2002) Function and behavior representation in conceptual mechanical design. AIEDAM 16:343–362

Deng YM, Tor SB, Britton GA (2000a) A dual-stage functional modelling framework with multi-level design knowledge for conceptual mechanical design. J Eng Des 11(4):347–375

Deng YM, Tor SB, Britton GA (2000b) Abstracting and exploring functional design information for conceptual mechanical product design. Eng Comput 16:36–52

Erden MS, Komoto H, Van Beek TJ, D'Amelio V, Echavarria E, Tomiyama T (2008) A review of function modelling: approaches and applications. AIEDAM 22:147–169

Goel AK, Stroelia E (1996) Functional device models and model-based diagnosis in adaptive design. AIADAM 10:355–370

Goel AK, Rugaber S, Vattam S (2009) Structure, behavior, and function of complex systems: the structure, behavior, and function modeling language. AIEDAM 23:23–35

Hirtz J, Stone RB, McAdams DA, Szykman S, Wood KL (2002) A functional basis for engineering design: reconciling and evolving previous efforts. Res Eng Des 13:65–82

Kitamura Y, Mizoguchi R (2003) Ontology-based description of functional design knowledge and its use in a functional way server. Expert Syst Appl 24:153–166

Kitamura Y, Mizoguchi R (2004) Ontology-based systematization of functional knowledge. J Eng Des 15(4):327–351

Kitamura Y, Koji Y, Mizoguchi R (2005/2006) An ontological model of device function: industrial deployment and lessons learned. Appl Ontol 1:237–262

Kitamura Y, Takafuji S, Mizoguchi R (2007) Towards a reference ontology for functional knowledge interoperability. In: Proceedings of the ASME 2007 international design engineering technical conferences & computers and information in engineering conference (IDETC/CIE), Las Vegas, 4–7 September 2007

Lind M (1994) Modeling goals and functions of complex industrial plants. Appl Artif Intell 8:259–283

Nagel RL, Vucovich JP, Stone RB, McAdams DA (2007a) Signal flow grammar from the functional basis. In: Proceedings of international conference on engineering design (ICED 07), Paris, 28–31 August 2007

Nagel RL, Stone RB, McAdams DA (2007b) A theory for the development of conceptual functional models for automation of manual processes. In: Proceedings of the ASME 2007 international design engineering technical conferences & computers and information in engineering conference (IDETC/CIE), Las Vegas, 4–7 September 2007

Ookubo M, Koji Y, Sasajima M, KitamuraY, Mizoguchi R (2007) Towards interoperability between functional taxonomies using an ontology-based mapping. In: Proceedings of international conference on engineering design (ICED 07), Paris, 28–31 August 2007

Otto K, Wood KL (1998) Product evolution: a reverse engineering and redesign methodology. Res Eng Design 10(4):226–243

Otto KN, Wood KL (2001) Product design: techniques in reverse engineering and new product development. Prentice Hall, Upper Saddle River

Pahl G, Beitz W (1988) Engineering design: a systematic approach. Springer, Berlin

Stone RB, Wood KL (2000) Development of a functional basis for design. J Mech Des 122:359–370

Stone RB, Wood KL, Crawford RH (1998) A heuristic method to identify modules from a functional description of a product. In: Proceedings of 1998 ASME design engineering technical conferences, Atlanta, 13–16 September 1998

Stone RB, Wood KL, Crawford RH (2000) Using quantitative functional models to develop product architectures. Des Stud 21:239–260

Stone RB, McAdams DA, Kayyalethekkel V (2004) A product architecture-based conceptual DFA technique. Des Stud 25:301–325

Szykman S, Racz JW, Sriram RD (1999) The representation of function in computer-based design. In: Proceedings of the 1999 ASME design engineering technical conferences, Las Vegas, 1999

Szykman S, Fenves SJ, Keirouz W, Shooter SB (2001) A foundation for interoperability in next-generation product development systems. Comput Aided Des 33:545–559

Umeda Y, Ishii M, Yoshioka M, Shimomura Y, Tomiyama T (1996) Supporting conceptual design based on the function-behavior-state modeller. AIEDAM 10:275–288

Van der Vegte W, Kitamura Y, Koji Y, Mizoguchi R (2004) Coping with unintended behavior of users and products: ontological modeling of product functionality and use. In: Proceedings of the ASME 2004 design engineering technical conferences & computers in engineering conference, Salt Lake City, 28 September–2 October 2004

Van Eck D, McAdams DA, Vermaas PE (2007) Functional decomposition in engineering: a survey. In: Proceedings of the ASME 2007 international design engineering technical conferences & computers and information in engineering conference (IDETC/CIE), Las Vegas, 4–7 September 2007

Zhang WY, Lin LF, Tong RF, Li X, Dong JX (2005) XB-FES/o: an ontology-based scheme for functional modelling in design for semantic web applications. In: Proceedings of the 9th international conference on computer supported cooperative work in design, vol 2, pp 1146–1151