AI EDAM Special Issue on Functional Descriptions

Making function modeling practically usable

Tetsuo Tomiyama, Delft University of Technology, Delft, The Netherlands (t.tomiyama@tudelft.nl)
Thom J. van Beek, Delft University of Technology, The Netherlands (thomvanbeek@gmail.com)
Andrés Alberto Alvarez Cabrera, Delft University of Technology*, Delft, The Netherlands (alvarezcabrera@yahoo.com)
Hitoshi Komoto, National Institute of Advanced Industrial Science and Technology*, Tsukuba, Japan (h.komoto@aist.go.jp)
Valentina D'Amelio, Delft University of Technology*, Delft, The Netherlands (valentina.damelio@gmail.com)

0. Abstract

Function modeling is considered potentially useful in various fields of engineering including engineering design. However, a close look at practices reveals that practitioners do not use function modeling so much, while in industry the concept of "function" frequently appears in many methods but not in a formal way. This paper first tries to understand why function modeling is not practically utilized in industry by analyzing usage cases of function. In the first place, the majority of product development is routine design or improvement design and does not require function level design, leading to less usage of function modeling. The paper then identified three problems that prevent function modeling from wider applications in practices, namely, practitioners' neglect of function modeling, the lack of function reasoning, and the complexity of function modeling. Finally, the paper proposed strategies to attack these problems and illustrated some research efforts to do so.

1. Introduction

If you are a researcher in design theory and methodology, in particular, function modeling, sometimes you are asked a question about the utility of the field: "How can I use your research results?†" Although some ask this purely out of curiosity, others try to imply that indeed design theory and methodology is useless, because people can design even without design theory and methodology. Others try to mean that it might be useful, but it is such a trivial area that there is no need to pay serious attention.

Unfortunately, this tendency exists in industry as well. At educational institutions, in machine design courses, students are taught systematic design methodologies (such as (Hubka & Eder, 1982; VDI, 1993; Pahl et al. 2007) and function modeling. Any design should begin with defining functional

* The work was done while the authors were working at Delft University of Technology as a PhD student.

† One of the authors was indeed asked this question during a conference in which his PhD student was presenting a theory about function modeling.

requirements, followed by establishing functional structure through functional analysis. Conceptual design is largely driven by functional requirements. By following such a design methodology, the student is expected to obtain at least good design, if not the best or innovative. These students are disappointed when they discover real practices in industry, because design methodologies are not used at all or function modeling is rarely used or even mentioned. Design practices are often intuitive and not systematic. Conceptual design is paid least attention and creative design is heavily relying only on brainstorming. Consequently, design outputs lack rational explanations and there is no guarantee that the "discovered" solution is "the best solution". This situation sounds like an exaggeration but more or less this is the observation of industrial practices made by the authors.

Typically industrial practitioners do not regard function modeling something very useful, particularly, for the purpose of design. They have been taught function modeling at school, so they know how to draw a function diagram but they never make one. However, this does not mean that the concept of function itself is useless for practicing designers and engineers in industry. There are a variety of engineering methods (many of which are related to quality, though) that require explicit representation of functions. These methods include QFD (Quality Function Deployment) (Mizuno & Akao, 1993), FMEA (Failure Mode Effect Analysis) (McDermott et al., 1996), value engineering (Younker, 2003), DSM (Design Structure Matrix) (Steward, 1981; Browning, 2001) to name a few. Function here means almost anything, ranging from "academic function" to physical behaviors to attributes, although almost all of these methods do not assume a special function modeling method but a classic, generic verb-noun pair form ("*to do something*"). As such, no coherent function modeling exists common to these engineering methods; in some cases a function model is a simple collection of statements in a natural language, and in other cases a graphical representation consisting of boxes and arrows.

Function modeling is a formal way to define and model functions. According to review articles on function modeling and its relevant fields (Umeda & Tomiyama, 1997; Erden et al., 2008; Borgo et al., 2009), there are three major different function modeling methods, *viz*., transformational methods (Chakrabartti & Bligh, 1996, 2001; Stone & Wood, 2000; Pahl et al. 2007), methods with physical behavior as medium including the Function-Behavior-State modeling (Umeda et al., 1990; Umeda et al., 1996) and the Function-Behavior-Structure modeling (Gero, 1990; Gero et al., 1992), and their variations (e.g., Deng, 2002), and state transition based methods (Goel, 1992; Goel & Stroulia, 1996; Goel et al., 2009). Functional ontology has been also intensively studied (Kitamura et al., 2002, 2004; Kitamura & Mizoguchi, 2004; Borgo et al., 2009; Carrara et al., 2011). Since this paper does not aim at giving detailed comparative descriptions about these three methods, interested readers are invited to refer to the review survey articles mentioned above. As suggested above, however, none of these modeling methods is practically used by industry in which, instead, function is defined in an *ad hoc* manner depending on the usage without any formal model.

This paper tries to understand why academically developed formal function modeling methods are not used but the concept of function itself is commonly used in practice[‡]. To do so, first it analyzes various

‡ Unfortunately, the discussion is mostly based on our observations and not based on any formal survey or investigation. However, it is not superficial, because authors were embedded within an actual product development environment and had discussions with team members including system architects and domain experts.

types of engineering methods that rely on the concept of function. The paper will identify four usage types of function, *viz*., to represent the purpose of the artifact, to explain the behavior, structure, or working principle of the target system, as a means to capture customer requirements, and as a means to illustrate the overview of the system. It will then try to understand problems associated with practical use of function modeling. Function modeling is not used in practice, primarily because truly innovative new design that requires function modeling rarely happens. In addition, practitioners do not simply believe that function modeling is useful, described functions cannot be fully utilized due to the lack of function reasoning, and as the size and complexity of the target system grows, the function model quickly "explodes" and cannot be dealt with easily. Finally the paper will propose countermeasures for these problems and introduce some attempts to attack these problems including our own work.

2. Usage of functional descriptions

Since the majority of textbooks on (systematic) design methodologies (Otto & Wood, 2001; Pahl et al. 2007; Ulrich & Eppinger, 2011) advocate conceptual design in which customer requirements are translated into functional structure and further into an embodied architecture, design researchers would like to believe that function is heavily used during design. This is not the case for most of product development activities in industry, however, because they are not new design but routine design or improvement design in which the product architecture is given or fixed. In contrast, when a completely new product is to be designed, function is supposed to play the key role during the conceptual design stage. However, due to increasing time pressure (for instance, time to market) there is a strong tendency of designers and engineers to move on to later stages of design as quickly as possible. As a consequence, unfortunately, insufficient time and effort is paid to conceptual design even for new design[§].

Additionally, it is well known that (systematic) function-driven design methodologies are not fully utilized in industrial environments anyway (Tomiyama et al., 2009). Even for new designs aiming at innovation, very few formal methods are used. Typically during conceptual design ideas are generated (almost solely) with brainstorming and once the most seemingly promising ideas are found, primarily these ideas are pursued and reflections for alternative solutions are seldom given. This suggests that there is a little chance that function based design is exercised in industrial practices, which could be another explanation for the lack of the use of formal function modeling methods. However, there are increasing cases in practitioners find useful to use such methods as Suh's axiomatic design (Suh, 1990) and TRIZ (Altshuller, 1984), both of which require functional descriptions (but in a classic verb-noun pair form).

On the other hand, both in literature and through observations, it is possible to find out the use of functional descriptions in a variety of engineering methods but non-function-based design contexts. The followings are non-exhaustive examples of such functional descriptions. These methods do not employ a specific function modeling method but a classic verb-noun pair form ("*to do something*") or sometimes

---

[§] It is not easy to find quantitative data to support this statement in literature. However, there is, for example, work done by Sobek and Jain (2007) on the relationship between the design quality and time and effort spent in different stages of design.

sentences in a natural language explaining such concepts as behaviors, working mechanisms, inputs and outputs.

Requirement descriptions: During the product definition stage (i.e., before conceptual design), documents that describe and analyze customer requirements are made. These customer requirements are broken down into technical specifications and stored and reviewed. However, they are not really used in the succeeding stages throughout the product development process, especially within the core development team. However, these documents are indeed used for inter-organization communication purposes with different organizations (for instance, between the main product development team and an external software development team) and for validation/verification and review purposes at formal occasions such as design reviews. In addition, these documents serve certification and quality control purposes.

QFD (Quality Function Deployment) (Mizuno & Akao, 1993): QFD is a method to reflect customer's requirements in technical functions and further down in technical specifications. All of these descriptions can include functional descriptions. By creating a house of quality, one tries to balance and prioritize these elements. In QFD, functions are used but these functions are not related to any technical functions used in, e.g., conceptual design.

Systems engineering: Systems level design follows conceptual design. Systems architecture is usually functional decomposition in which subsystems are defined and grouped according to their functions. In other words, an appropriate function model helps practitioners to understand the architecture, divided tasks, and eventually points to improve (Alvarez Cabrera et al., 2011). Methods to overview systems architecture of complex systems were developed (Sobek & Smalley, 2007) and (Borches Juzgado, 2010) both using a schematic representation on an A3 sheet. Also, systems engineering documents are full of behavior level information which is connected to functions (for instance, imagine functions in in SysML diagrams (OMG, 2010) although functional descriptions in SysML are a bundled set of input and output parameters and their transformation).

PLM (Product Lifecycle Management): In another company that involves multi-disciplinary products, they create documents containing functional and behavioral descriptions stored in a knowledge management system of PLM, but here functions are common denominator for elements in different domains. For instance, a "conveyor" has a "controller" to control a "motor". The controller was a part of system achieving functionality of transport and three separate depositories in mechanical (3D geometric model), software (a piece of controller code), and control domains (systems dynamics model representing differential equations). Here a function is rather an identifier.

Quality control (for example ISO 9000 series) (ISO, 2008): Quality control in product development basically requests to document all processes. Within these processes, documents that describe functional requirements are made following predefined templates. These documents are used for the validation purposes during the development process and for certification after the development.

Value engineering: In value engineering (Younker, 2003), value is defined by function/cost, so in order to increase value, functions must be increased or costs decreased. It is a method to achieve higher quality and lower costs. Functions are represented in the classic verb-noun pair form and should be evaluated against requirements.

FMEA (Failure Mode Effect Analysis) (McDermott et al., 1996): To increase reliability of the product, FMEA is a frequently exercised technique. Failures are defined as malfunctioning of the product, and as such FMEA requires first understanding the behavior and purpose of the system. FMEA uses function descriptions but usually in a classic verb-noun pair form.

Behind these usage examples of functional descriptions, one can identify four major purposes. These usage types may overlap each other.

(Umeda & Tomiyama, 1997; Erden et al., 2008; Borgo et al., 2009), there are three major different function modeling methods, *viz*., transformational methods (Chakrabartti & Bligh, 1996, 2001; Stone & Wood, 2000; Pahl et al. 2007), methods with physical behavior as medium including the Function-Behavior-State modeling and the Function-Behavior-Structure modeling, and their variations (e.g., Deng, 2002), and state transition based methods (Goel, 1992; Goel & Stroulia, 1996; Goel et al., 2009). Functional ontology has been also intensively studied (Kitamura et al., 2002, 2004; Kitamura & Mizoguchi, 2004; Borgo et al., 2009; Carrara et al., 2011).

(1) To represent the purpose of the artifact: This type is used during function-based conceptual design as a driving force of design (Umeda et al., 1990; Gero, 1990; Gero et al., 1992; Umeda et al., 1996; Chakrabartti & Bligh, 1996, 2001; Stone & Wood, 2000; Deng, 2002; Pahl et al. 2007). The function in FMEA also belongs to this type. In many cases, these can eventually be associated with behaviors, physical principles, and attributes. Formal function modeling methods can be used here.
(2) To explain the behavior, structure, or working principle of the target system: This type aims at giving an explanation (or understanding) of the artifact (for communication) (in addition to those authors in (1), Kitamura et al., 2002, 2004; Kitamura & Mizoguchi, 2004; Borgo et al., 2009; Carrara et al., 2011). For this purpose, formal function modeling and operation methods are assumed.
(3) As a means to capture customer requirements: This is obvious in the examples of requirement descriptions, QFD, and value engineering. For these, functional descriptions are often given in the verb-noun pair form or even in a natural language sentence.
(4) As a means to illustrate the overview of the system: This is perhaps a subcategory of (2), but for instance, in systems engineering, systems architecture can be often explained through functional decomposition (Borches Juzgado, 2010; Alvarez Cabrera et al., 2011). The usage example in PLM (Product Lifecycle Modeling) belongs to this category as well. Here, a function is a common denominator that connects chunks of information that belong to different fields (or views) during communication among stakeholders.

3. Why functional modeling is not used in industry?

Practitioners do not rely on functional descriptions as a driving force for conceptual design. The followings are possible reasons why they don't.

(1) *"Never used it" syndrome* (or simple neglect): Practitioners believe that they know how to model and

represent functions, because it looks very easy or at least not difficult. They have seen or heard of function modeling but never received any formal education or training about a formal method together with its potential powerful applications (for instance, functional analysis with respect to quality, cost, and architecture). Perhaps, they have never explored beyond "transformational boxes" or "to do something" verb-noun pairs. So, they know function modeling only superficially and cannot believe that it would bring in useful results.

(2) *"No added value" syndrome*: Practitioners don't feel like going deeper than what they (must) do now. For instance, if documenting functional requirements is needed only for the certification purpose, they do not add more information than necessary and will not explore other applications. They feel their time and effort to complete function diagrams were wasted, if there is no practical added value of the diagrams. This simply means even if one builds a function model, it becomes immediately useless.

(3) *"Not practical" syndrome*: For practitioners, academically developed methods are too formal, abstract, and far away from real products missing direct connections with any other information more frequently used (e.g., 3D geometric models). In addition, because these modeling methods were not seriously used in industry, they have never been improved to become useful and practical. This leads to another problem that since a function model of even a simple mechanism may explode if a complete picture of the entire artifact is necessary, because potentially it can contain a number components, relationships, and other types information that are interrelated (see Figure 1 illustrating an FBS diagram). Without a professionally developed tool that can handle so much information and proper training of the designer, practically functional diagram cannot fulfill the explanation or overview purposes. Additionally, it must be emphasized that often an appropriate function model helps practitioners to understand the architecture, divided tasks, and eventually points to improve. However, they also see too many new ideas. Without an appropriate method to reduce those generated ideas, the whole function modeling exercise often diverges rather than converges.

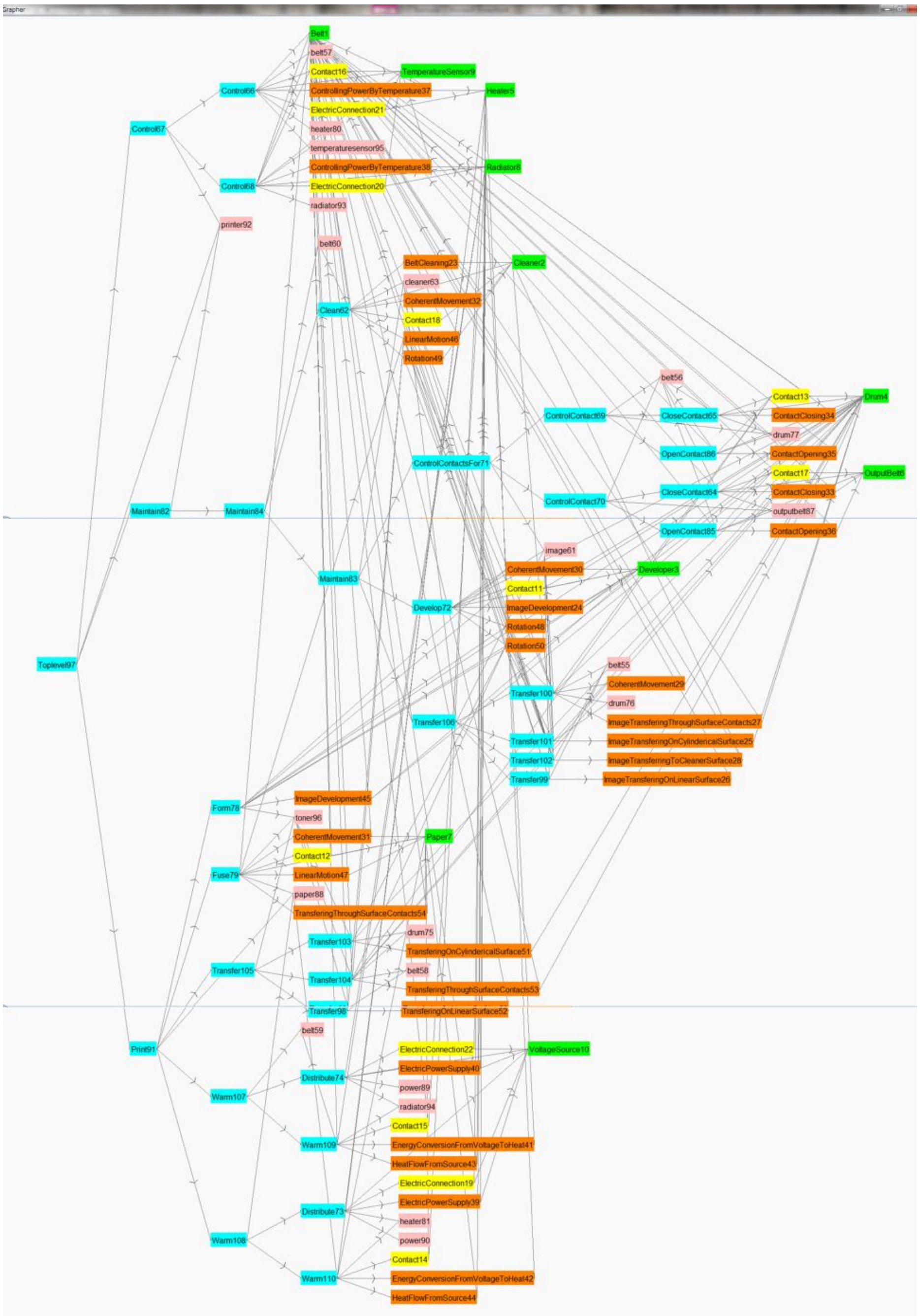

Figure 1. An FBS model of a simple printer mechanism (Blue boxes are functions, orange are physical phenomena, green are entities and yellow are relationships among entities)

4. Three possible directions to make function modeling more practical

Corresponding to the three syndromes listed above, we are now able to identify three possible directions for future development to make function modeling practically usable.

One way to tackle the "never used it" syndrome would be to demonstrate that truly creative design becomes systematically feasible with deep functional level analysis. Related to this, Tomiyama and Breedveld (Tomiyama et al., 2010; Breedveld 2011) developed a method called ACRREx (Abstracting, Categorizing, Reflecting, Reformulating and Extending) to generate creative solutions. In this method, the designer first needs to define a problem space through functional analysis and then to find "categorizations" (or "coordinates") that define the problem space (in Table 1 below, power source and the number of wheels) by abstracting essential characteristics that differentiate this space. Practically, this can be performed first by creating a morphological table built systematically or intuitively (Pahl et al., 2007) and second by systematically expanding the table with as many elements as possible ("reflecting", "reformulating" and "extending"). In Table 2 below, the designer realizes that in fact the number of wheels can be any positive number.

Table 1. First Attempt to Structure Wheeled Vehicles (Tomiyama et al., 2010)

	2 wheels	**4 wheels**
Human powered	Bike	**?**
Motorized	**?**	Car

Table 2. Completing the Structure of Wheeled Vehicles (Tomiyama et al., 2010)

	Number of wheels					
	1	**2**	**3**	**4**	***n***	**∞**
Human powered	unicycle	bike	tricycle	kart	**?**	**?**
Motorized	motorized unicycle	motorbike	motorized tri-cycle	car	large truck	military tank

Against the second "no added value" syndrome, we argue that primarily the lack of function reasoning to generate added value (after a function model is created) is the fundamental cause for the unpopularity of function modeling, rather than the lack of unified function modeling method. The current research trends pay too much attention to modeling and representation of function. Therefore, it is critical to demonstrate that functional modeling can indeed generate added value in design activities by deriving useful knowledge through function reasoning (Far & Elamy, 2005) that include the followings.

- Functional level simulation for validation: During systems architecting, validation needs to be performed quantitatively but this implies that validation can be performed only after some progresses are made. Function level (or qualitative) simulation would improve the efficiency of the whole product development process, but such a technique applicable to a wide range of models does not exist yet despite the progress in qualitative physics (Price et al., 2006). Instead, identifying overlooked design failures would be helpful at least to give warnings to designers for potential design failures (D'Amelio et al., 2011).
- Function redundancy design (Tomiyama et al., 1993; Umeda et al., 1994): The FBS (Function-Behavior-State) modeling was originally developed to look for redundant functions to improve "fault tolerance" and to design a "function-redundant type self-maintenance machine".
- Identification of latent functions: This might be useful to find out redundant functions or excess functions. This becomes possible with the functional level simulation.
- Function level techniques for systems architecting: For instance, Stone et al. (2000) demonstrated that it is possible to generate product architecture from a functional diagram. The authors' group also developed some techniques and tools addressing this topic (Komoto & Tomiyama, 2010, 2011). These

techniques could facilitate and improve "system-level design" (Sobek, 2006).

•

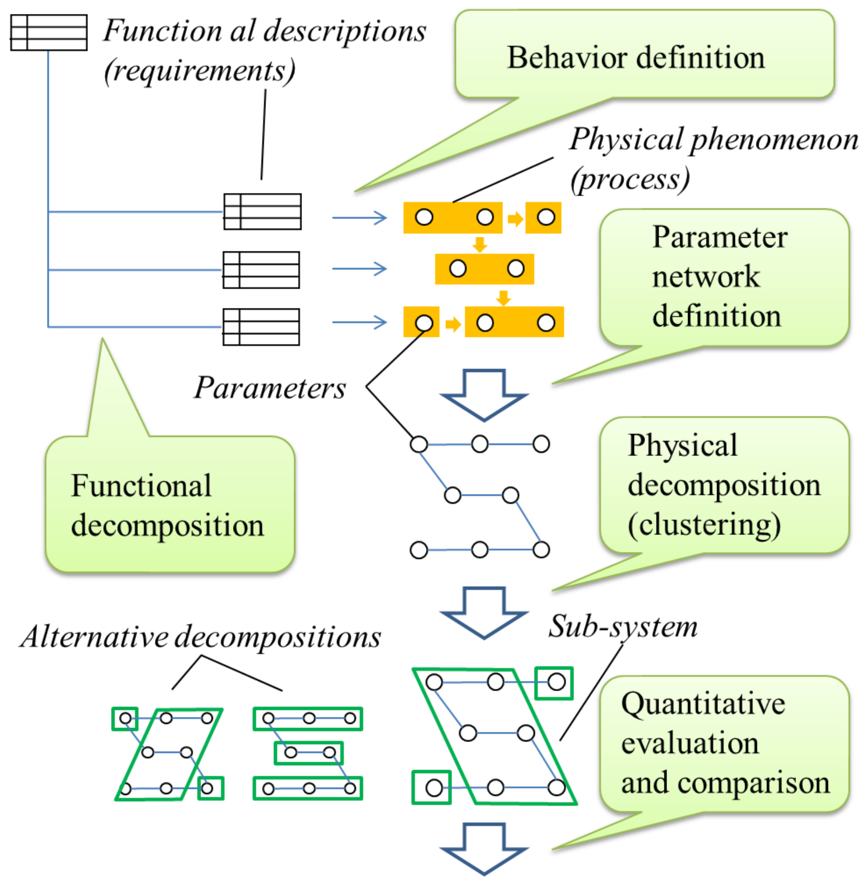

Figure 2. Systems Architecting Process

- Komoto and Tomiyama (2010, 2011) demonstrated that a computer-based tool can support the decomposition process during systems architecting (Figure 2). First, physical phenomena that embodies the required functions are identified from functional descriptions in the FBS (function-behavior-state) format (Umeda et al. 1996). These physical processes contain parameters and by clustering these parameters, different decompositions are systematically generated. Subsequent quantitative performance evaluation facilitates to choose the best architecture.
- Alvares Cabarera et al. (2011) developed a method and a computer tool (AM tool) for architecture-centric model-based product development. Architecture defines subsystems and their relationships among them and system architecting involves (hierarchical) decomposition, behavior definition of subsystems, and interface definition. Functions give an overview of subsystems (behaviors) and form the basis for cross-disciplinary communication (Figure 3). The tool can connect to further development stage, such as performance analysis as well as control software generation (Alvares Cabarera et al., 2010).
- Other possibilities would be connecting functional descriptions with user level upfront information such as workflow that would eventually guarantee and increase the usability of the

product in connection with functions (van Beek & Tomiyama, 2011). From functional descriptions, it is possible to generate subsystems interface definitions as DSM (Design Structure Matrix).

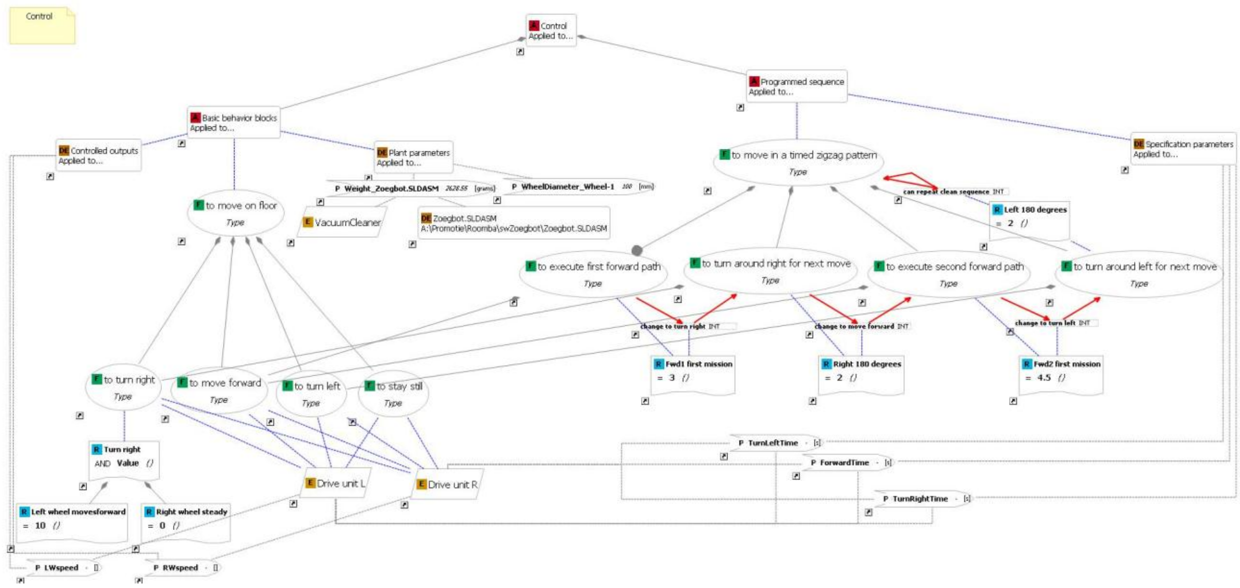

Figure 3. Modeling an Autonomous Vehicle in the AM Tool

Against the third "not practical" syndrome, countermeasures include the development of easy-to-use tools and training methods that can still handle reasonably complicated function structure (i.e., not a toy problem). Ideally, these tools and techniques must be extremely easy to use without even training. Since function level design can involve true end users (e.g., in case of medical equipment, doctors and nurses, not necessarily designers and engineers), tools like the FBS modeler (Figure 1) are too difficult. From this point of view, function level design should not involve function modeling but much easier ways of describing their needs, wishes, and constraints, such as the workflow modeling mentioned above (van Beek & Tomiyama, 2011).

Even for systems architects, engineers, and designers who have to deal with function modeling, the tool must be extremely easy to operate. In addition, the tool should be equipped with knowledge bases that can be professionally useful in function modeling. It must be capable of dealing with large amount of information (e.g., hierarchical treatment), versioning, and multiple user accesses.

5. Conclusions

This paper began with a proposition that function modeling is primarily researched by academics and taught at educational institutions but in reality practitioners don't use it. The paper analyzed this situation and found that, while formal function modeling methods are not used, the concept of function itself appears at every corner of engineering in a very simple form (such as the classic verb-noun form or sentences in a natural language). One major reason of function modeling being not used is that the majority of product development is routine design or improvement design and does not require function level design in the first place. Another reason is that practitioners don't recognize very well benefits of function modeling when applied to (new) design.

To go deeper into this observation, the paper analyzed various types of engineering methods that rely on the concept of function. The paper identified four usage types of function, *viz.*, to represent the purpose of the artifact, to explain the behavior, structure, or working principle of the target system, as a means to capture customer requirements, and as a means to illustrate the overview of the system. In all these usage types, we could identify practical problems why function modeling is not used in practice. These include: (1) Practitioners do not simply believe the usefulness of function modeling, (2) the lack of function reasoning, and (3) the explosion of the model as the size and complexity increase.

At the end, the paper proposed strategies about how to attack these problems. Against the lack of belief in utility of function modeling, one needs to demonstrate creative design generated with deep functional level analysis. Against the second lack of functional reasoning, we simply need to develop various methods to increase added value of function modeling. Among others, functional level simulation for validation and functional methods for architecting seem promising. Against the third practicality (complexity) problem, the development of easy-to-use tools and training methods is a must. These tools should be equipped with a professional level user interface and operational capabilities In addition, for real end users, we may think about replacement of function models with much easier models.

Finally we must conclude that function modeling itself is not a big issue. Bigger issues are how to advocate engineering design methods combined with function modeling and how to make better use of functional descriptions. To achieve this, we recommend three strategies: Show the usefulness of function modeling, derive useful information from function modeling, and develop practically usable professional tools.

6. Acknowledgement

The authors gratefully acknowledge the following projects that supported the part of the research work. "Automatic Generation of Control Software for Mechatronics Systems" (supported by the Innovation-Oriented Research Programme "Integral Product Creation and Realization (IOP IPCR)" of the Dutch Ministry of Economic Affairs, Agriculture and Innovation), "Darwin" and "Octopus" (carried out under the responsibility of the Embedded Systems Institute in Eindhoven and partially supported by the same ministry under the BSIK program). Philips Healthcare and Océ Technologies were industrial partners of Darwin and Octopus projects, respectively.

7. References

Altshuller, G.S. (1984). *Creativity as an Exact Science*. New York: Gordon and Breach Science Publishers.

Alvarez Cabrera, A.A., Foeken, M.J., Tekin, O.A., Woestenenk, K., Erden, M.S., De Schutter, B., van Tooren, M.J.L., Babuška, R., van Houten, F.J.A.M., & Tomiyama, T. (2010). Towards automation of control software: A review of challenges in mechatronic design. *Mechatronics* 20(8), 876–886.

Alvarez Cabrera, A.A., Woestenenk, K., & Tomiyama, T. (2011). An architecture model to support cooperative design for mechatronic products: A control design case. *Mechatronics* 21(3), 534–547.

van Beek, T.J., & Tomiyama, T. (2011). Workflow Modelling of Intended System Use. In *Views on Evolvability of Embedded Systems*, (van de Laar, P., & Punter, T. Eds.), Springer, pp. 153–170.

Borches Juzgado, P.D. (2010). *A3 Architecture Overviews.* PhD Thesis. Enschede, The Netherlands: Twente University.

Borgo, S., Carrara, M., Garbacz, P., & Vermaas, P.E. (2009). A formal ontological perspective on the behaviors and functions of technical artifacts. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 23(1), 3-21.

Breedveld, P., Herder, J.L., & Tomiyama, T. (2011). Teaching Creativity in Mechanical Design. In *Proceedings of IASDR2011, the 4th World Conference on Design Research,* (Roozenburg, N.F.M., Chen, L.L., & Stappers, P.J. Eds.), *Delft, the Netherlands*, pp. 1–10, *31 October - 4 November*.

Browning, T.R. (2001). Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *IEEE Transactions on Engineering Management* 48(3), 292–306.

Carrara, M., Garbacz, P., & Vermaas, P.E. (2011). If engineering function is a family resemblance concept: Assessing three formalization strategies. *Applied Ontology* 6(2), 141–163.

Chakrabarti, A., & Bligh, T.P. (1996). An approach to functional synthesis of mechanical design concepts: theory, applications, and emerging research issues. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 10(4), 313–331.

Chakrabarti, A., & Bligh, T.P. (2001). A scheme for functional reasoning in conceptual design. *Design Studies* 22(6), 493–517.

D’Amelio, V., Chmarra, M.K., & Tomiyama, T. (2011). Early design interference detection based on qualitative physics. *Research in Engineering Design* 22(4), 223–243.

Deng, Y.M. (2002). Function and behavior representation in conceptual mechanical design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* (16)5, 343–362.

Erden, M.S., Komoto, H., van Beek, T.J., D’Amelio, V., Echavarría, E., & Tomiyama, T. (2008). A review of function modeling: Approaches and applications. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 22(2), 147–169.

Far, B.H., & Elamy, A.H. (2005). Functional reasoning theories: Problems and perspectives. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 19(2), 75–88.

Gero, J. (1990). Design Prototypes: A Knowledge Representation Schema for Design. *AI MAGAZINE* 11(4), 26–36.

Gero, J.S., Tham, K., & Lee, H. (1992). Behaviour: a link between function and structure in design. *IFIP Transactions B: Applications in Technology, Intelligent Computer Aided Design*, B-4, 193–225.

Goel, A. (1992). Representation of Design Functions in Experience-Based Design. *IFIP Transactions B: Applications in Technology, Intelligent Computer Aided Design*, B-4, 283-308.

Goel, A.K., Rugaber, S., & Vattam, S. (2009). Structure, behavior, and function of complex systems: The structure, behavior, and function modeling language. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 23(1), 23–35.

Goel A.K., & Stroulia, E. (1996). Functional device models and model-based diagnosis in adaptive design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 10(4), 355–370.

Hubka, V., & Eder, W.E., 1982, *Principles of Engineering Design*. London: Butterworths Scientific.

ISO/TC176, ISO 9001:2008, *Quality management systems – Requirements*, 2008.

Kitamura, Y., Kashiwase, M. Fuse, M., & Mizoguchi, R. (2004). Deployment of an ontological framework of functional design knowledge. *Advanced Engineering Informatics* 18(2), 115–127.

Kitamura, Y., & Mizoguchi, R. (2004). Ontology-based systematization of functional knowledge. *Journal of Engineering Design* 15(4), 327–351.

Kitamura, Y., Sano, T., Namba, K., & Mizoguchi, R. (2002). A functional concept ontology and its application to automatic identification of functional structures. *Advanced Engineering Informatics* 16(2), 145–163.

Komoto, H., & Tomiyama, T. (2010). A system architecting tool for mechatronic systems design. *CIRP Annals–Manufacturing Technology* 59(1), 171–174.

Komoto, H., & Tomiyama, T. (2011). Multi-disciplinary system decomposition of complex mechatronics systems. *CIRP Annals–Manufacturing Technology* 60(1), 191–194.

McDermott, R.E., Mikulak, R.J., & Beauregard, M.R. (1996). *The Basics of FMEA*. New York: Productivity Press.

Mizuno, S., & Akao, Y. (1993). *QFD: The Customer-Driven Approach to Quality Planning & Deployment*. Tokyo: Asian Productivity Organization.

OMG (Object Management Group) (2010). *OMG Systems Modeling Language (OMG SysMLTM)* Version 1.2, OMG Document Number: formal/2010-06-02, URL: http://www.omg.org/spec/SysML/1.2/ (accessed on 13 April 2012).

Otto, K., & Wood, K. (2001). *Product Design, Techniques in Reverse Engineering and New Product Development*. Upper Saddle River, NJ: Prentice Hall.

Pahl, G., Beitz, W., Feldhusen, J., & Grote, K.-H. (2007). *Engineering Design–A Systematic Approach, Third Edition*. (Wallace, K., & Blessing, L. Translated and Eds.), Berlin: Springer.

Price, C.J., Travé-Massuyès, L., Milne, R., Ironi, L., Forbus, K., Bredeweg, B., Lee, M.H., Struss, P., Snooke, N., Lucas, P., Cavazza, M., & Coghill, G.M. (2006). Qualitative futures. *Knowledge Engineering Review* 21(4), 317-334.

Sobek, D.K. (2006). System-Level Design: A Missing Link? *International Journal of Engineering Education* 22(3), 533–539.

Sobek, D.K. II, & Jain, V.K. (2007). Relating Design Process to Quality: A Virtual Design of Experiments Approach. *ASME Journal of Mechanical Design* 129(5), 483-490.

Sobek, D.K. II, & Smalley, A. (2008). *Understanding A3 Thinking: A Critical Component of Toyota's PDCA Management System*. New York: Productivity Press.

Steward, D.V. (1981). The Design structure system: A method for managing the design of complex systems. *IEEE Transactions on Engineering Management* 28(3), 71–74.

Stone, R.B., & Wood, K.L. (2000). Development of a functional basis for design. *ASME Journal of Mechanical Design* 122, 359-370.

Stone, R.B., Wood, K.L., & Crawford, R.H. (2000). A heuristic method for identifying modules for product architectures. *Design Studies* 21(1), 5–31.

Suh, N.P. (1990). The Principles of Design. Oxford: Oxford University Press.

Tomiyama, T., Breedveld, P., & Birkhofer, H. (2010). Teaching Creative Design by Integrating General Design Theory and the Pahl and Beitz Methodology. In *Proc. of the ASME 2010 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference,* Montreal, Quebec, Canada, August 15-18.

Tomiyama, T., Gu, P., Jin, Y., Lutters, Y., Kind, C., & Kimura, F. (2009). Design methodologies: Industrial and educational applications. *CIRP Annals–Manufacturing Technology* 58(2), 543–565.

Tomiyama, T., Umeda, Y., & Yoshikawa, H. (1993). A CAD for functional design. *CIRP Annals–Manufacturing Technology* 42(1), 143–146.

Ulrich, K.T., & Eppinger, S.D. (2011). *Product design and development, Fifth Edition*, New York: McGraw-Hill.

Umeda, Y., Takeda, H., Tomiyama, T., & Yoshikawa, H. (1990). Function, behaviour, and structure. In *Applications of artificial intelligence in engineering V, Proceedings of the Fifth International Conference,* vol. 1, (Gero, J.S. Ed.), pp. 177–194, Southampton & Boston: Computational Mechanics Publications, *July 1990*.

Umeda, Y., Ishii, M., Yoshioka, M., Shimomura, Y., & Tomiyama, T. (1996). Supporting conceptual design based on the function-behavior-state modeler. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 10(4), 275–288.

Umeda, Y., Tomiyama, T., Yoshikawa, H., & Shimomura, Y. (1994). Using Functional Maintenance to Improve Fault-Tolerance. *IEEE Expert* 9(3), 25–31.

Umeda, Y., & Tomiyama, T. (1997). Functional reasoning in design. *IEEE Expert* 12(2), 42–48.

VDI, (1993). *VDI 2221: Systematic approach to the development and design of technical systems and products.* Berlin: Beuth Verlag.

Younker, D.L. (2003). *Value Engineering: Analysis And Methodology*. CRC Press.