Artificial Intelligence for Engineering Design, Analysis and Manufacturing (2013), *27*, 249–257.

Cambridge University Press 2013 0890-0604/13 $25.00
doi:10.1017/S0890060413000255

RESPONSE PAPER

The function analysis diagram: Intended benefits and coexistence with other functional models

MARCO AURISICCHIO,[1] ROB BRACEWELL,[2] AND GARETH ARMSTRONG[3]

[1]Design Engineering Group, Mechanical Engineering Department, Imperial College London, London, United Kingdom
[2]Engineering Design Centre, Engineering Department, University of Cambridge, Cambridge, United Kingdom
[3]Design Systems Engineering, Rolls-Royce plc, Derby, United Kingdom

(Received March 1, 2012; Accepted March 1, 2013)

Abstract

Understanding product functions is a key aspect of the work undertaken by engineers involved in complex system design. The support offered to these engineers by existing modeling tools such as the *function tree* and the *function structure* is limited because they are not intuitive and do not scale well to deal with real-world engineering problems. A research collaboration between two universities and a major power system company in the aerospace domain has allowed the authors to further develop a method for function analysis known as function analysis diagram that was already in use by line engineers. The capability to generate and edit these diagrams was implemented in the Decision Rationale editor, a software tool for capturing design rationale. This article presents the intended benefits of the method and justifies them using an engineering case study. The results of the research have shown that the function analysis diagram method has a simple notation, permits the modeling of product functions together with structure, allows the generation of rich and accurate descriptions of product functionality, is useful to work with variant and adaptive design tasks, and can coexist with other functional modeling methods.

Keywords: Design Rationale; Functional Analysis; Functional Modeling; TRIZ

1. INTRODUCTION

The engineering design process is a key part of new product introduction projects. It can be viewed as a human problem-solving activity in which customer needs and functional specifications are translated into a specification of the final product. With the increasing complexity of products and processes, there is a growing demand on engineering designers to understand and control the complex relationships between the required behavior and the physical structure of the design object. The provision of effective support in understanding useful and harmful functional relationships is, therefore, a fundamental aspect in the delivery of new product introduction projects.

Design support tools generally assist designers in the generation of models and abstractions, and are required because of cognitive limitations and problem complexity. In industry, the existing tools are predominantly quantitative in nature and tend to focus on the later phases of the design process [e.g., dynamic models, finite element analysis, computational fluid dynamics, and computer-aided design (CAD)]. In the initial design phases, qualitative tools are available [e.g., brainstorming, quality function deployment (QFD), design structure matrices, Pugh matrices, morphological analysis, and TRIZ]. However, the extent to which and the rigor with which they are used varies from industry to industry, and it is never as good as with quantitative design tools (Lopez-Mesa & Bylund, 2011). Qualitative tools tend to place emphasis either on the stimulation of creativity or on the structure and analysis of design information. A subset of them is able to assist designers and engineers in some form of functional reasoning (e.g., QFD, design structure matrices, morphological analysis, and TRIZ). Functional reasoning has a very important role in ensuring design quality and product innovativeness (Miles, 1972; Umeda & Tomiyama, 1997). The practical importance of functional reasoning in engineering design is also demonstrated by the fact that popular assembly and reliability methods rely on functional models (e.g., design for assembly, failure mode effect analysis, and value engineering). Function is an important concept in design, and it is at the base of numerous theories and models. Research on functional

Reprint requests to: Marco Aurisicchio, Design Engineering Group, Mechanical Engineering Department, Imperial College London, London, United Kingdom. E-mail: m.aurisicchio@imperial.ac.uk

analysis has contributed to the development of several ways of representing functions to support engineers in their tasks: function tree (Value Analysis Incorporated, 1993), data flow diagram (Yourdon, 1989), and function structure (Pahl et al., 2007). However, it is commonly reported that engineering designers make little use of such tools in industry today (Araujo et al., 1996; Whybrew et al., 2001; Lopez-Mesa & Bylund, 2011).

This research investigates computer-based modeling of functional interactions in engineered systems using the function analysis diagram (FAD) as implemented in the Decision Rationale editor (DRed; Aurisicchio et al., 2012). The FAD method, originally published as part of a patent application filed by the TRIZ vendor Invention Machine Corporation (Devoino et al., 1997), is a form-dependent product representation, which has received little attention compared to mainstream methods. Most probably a reason for the relative neglect of this approach by academia is that its reliance on product parts to model functions has made it unsuitable to achieve the objective, common to most researchers, of developing a form-independent product representation to support original design. The principal aim of the research reported in this paper is to develop a theoretical understanding of the FAD representation by researching its modeling characteristics and distinguishing it from other functional models. For this purpose, a FAD model of a centrifugal water pump is illustrated and compared to a function structure model of the same device (Aurisicchio et al., 2012; Eckert, 2013). Overall, the results of this research suggest that the FAD model is intuitive and easy to generate. The research also shows that FAD can coexist with the function structure and other modeling methods because they support complementary engineering design tasks (Vermaas, 2013).

2. BACKGROUND

The concept of function is rooted in the theoretical foundations of the most prevalent schools of thought in design, including design as a "science of the artificial" (Simon, 1996), design as a "structured systematic activity" (Pahl et al., 2007), and axiomatic design (Suh, 2001). Despite the centrality of function to design, there is no stable or generally accepted meaning of function available (Crilly, 2010; Vermaas, 2013). In engineering design, most definitions of function often share a notion of performing a transformative operation (e.g., "a function is the general input/output relationship of a system whose purpose is to perform a task"; Pahl et al., 2007). This definition generally raises the criticism that it leaves many nontransformative functions unaccounted for (e.g., retaining, guiding, and supporting). A function is, often, expressed by a statement including an "active verb," which represents the **action** performed, and a "noun," which represents the **object** upon which such action impacts [e.g., "*seals* (verb) *the fluid* (object)"]. The **subject** of the statement is the product element performing the function also expressed as a "noun" [e.g., "*the piston ring* (subject) *seals* (verb) *the fluid* (object)"].

Research on functional analysis has produced several representational formalisms. Examples of these are the function tree (Value Analysis Incorporated, 1993), the data flow diagram (Yourdon, 1989), and the function structure (Pahl et al., 2007). They often consist of specifying the overall function of the product under analysis and then of determining and mapping the subfunctions involved. However, the concept of function used varies from model to model, and the representations differ because of their components and the way they are organized. The next two sections review formalisms to represent product functionality distinguishing between form-independent and form-dependent models.

2.1. Form-independent functional models

The function tree is a simple method for functional analysis, which produces a form-independent model. There are two main methods to develop a function tree. The first is the functional analysis system technique (Value Analysis Incorporated, 1993), a top-down approach, in which functions may be generated through a brainstorming session. In the tree generated using this approach, there is a how–why relationship between a function and its subfunctions. For example, referring to the design of a potato peeler, one can ask "How does the potato peeler remove skin?" and answer "By limiting the depth of cut." The second method is the subtract and operate procedure, a bottom-up approach whose underlying assumption is that a product design or the actual physical model already exists (Lefever & Wood, 1996). A key problem of the function tree method is that the representation is not suitable for capturing the network of interconnected functional relationships present in most systems. A more sophisticated type of function tree was reported in (Kitamura et al., 2002). Based on a functional concept ontology, this representation differs from the conventional function tree because of the addition of transversal relationships between functions belonging to different branches and the use of product structure to label the nodes of the hierarchy.

More robust and complete functional models can be generated using approaches like the data flow diagram (also known as function flow diagram) and the function structure. These methods, originating from research work in system theory (Bertalanffy, 1969), are conceptually very similar and produce form-independent models. They both aim at modeling functions on flows and differ mainly because of secondary components of the notations. Other approaches for functional modeling stemming from research in system theory are the bond graph for energy transformation functionality and the Petri net for specifying how a system should respond to asynchronous events. An example of their application can be seen in the Schemebuilder project (Bracewell & Sharpe, 1996).

The data flow diagram has a representation in which a circle is used to represent a function (sometimes referred to as a process or transformation) and an arrow to represent a flow (Yourdon, 1989). It started as a tool to model information-processing systems in software engineering and has subsequently found application in system engineering to model complex systems. Since its introduction, modeling methods have evolved and examples of tools available to software and system engineers are the Unified Modeling Language, the System Modeling Language, and the object–process methodology (Dori, 2002).

The function structure is the standard convention used in academia over the last 30 years (Hubka & Eder, 1984; Ullman, 1992; Ulrich & Eppinger, 1995; Otto & Wood, 2001; Pahl et al., 2007). The method consists of drawing a flowchart with blocks describing the subfunctions of a product connected by arrows (in input and output) describing flows of matter, energy, and signals. As intended by the original proponents and used by its early researchers, the function structure is a method to capture a mesh of functional relationships (Hubka & Eder, 1984; Pahl et al., 2007). However, it is believed that in an attempt to reduce the complexity of the method, recent applications (Otto & Wood, 2001) have generally produced meshes with low internode connectivity resulting in predominantly left to right linear chains of functions.

The lack of precise definitions for subfunctions and flows has spurred research into the development of a high-level design language (sometimes called a vocabulary or taxonomy) to describe product function and thus enable a systematic approach to functional modeling. After the appearance of a range of initial functional taxonomies (Collins et al., 1976; Hundal, 1990; Little et al., 1997; Szykman et al., 1999; Stone & Wood, 2000; Pahl et al., 2007), research work has focused on reconciling previous efforts in what was termed the reconciled functional basis (RFB; Hirtz et al., 2002). The RFB is a controlled vocabulary containing *54 function verbs* and *45 flows or objects of action* arranged in a three-level hierarchy. The RFB is intended to be broad enough to span the entire mechanical design space while not being repetitive. Researchers in the field of functional analysis argue that functional modeling through the function structure and the RFB increases the clarity of the design problem by revealing functional and flow dependencies and tracking input and output flows (Bryant et al., 2005). Other applications of the function structure are as a foundation for design repositories; as support for new knowledge-based design methods such as design by analogy, as a design for manufacturing and product architecture; and teaching tool for design education and training (Hirtz et al., 2002).

Despite the extensive exploration of the function structure and the RFB, their benefits to engineering design practice remain questionable. The function structure, as used in most recent applications (Otto & Wood, 2001), produces a mesh with low internode connectivity, and therefore like the function tree, it is unsuitable for capturing networks of interconnected functional relationships. Additional limitations are the long learning time needed to master its rules and conventions as well as the abstract nature of the representation. Form-independent functional representations pose challenges to engineers because their natural way of working entails shifting between function and form-based reasoning.

The RFB has also started to attract academic criticism. Due to the restricted number of entries in the database, a lack of precision and completeness in the description of the abstract structure of a device was identified (Fantoni et al., 2009). The primary practical downside of this aspect is that designers are forced to think in very highly abstract terms. In addition, the verbs of the RFB were found to have several ambiguities and repetitions, which it was argued are unavoidable with a treelike structure (Bonaccorsi et al., 2009). In order to address some of these issues, the authors of these criticisms have proposed a new functional base architecture whose main characteristic is that the vertical levels of generality are based on physical, chemical, and logical laws (Fantoni et al., 2009).

2.2. Form-dependent functional models

Not all research has focused on form-independent modeling. There is a stream of research on functional modeling that has developed form-dependent representations, and this work comes from the artificial intelligence in design community. Examples are the structure–behavior–function model (Bhatta et al., 1996) and causal functional representational language (Iwasaki et al., 1993). Due to their complexity and ambitious goals to support sophisticated computational reasoning, none of these has developed into a widely used method.

To this group belongs also the function analysis diagram. The diagram consists of drawing a mesh with *blocks* used to represent product structure, users, or other resources, and *relations* in the form of an arrow with a label (strictly a relation node with one or more arrows in and out) used to represent either useful or harmful actions. A FAD, unlike the function tree and the function structure, represents functions together with the physical elements of a product. Function structure modeling uses blocks to represent functions and flow arrows for energy, materials, and information; but FAD modeling uses blocks to represent product structure, including carriers of energy such as wires and shafts, and volumes of material either internal to or in transfer between systems. Labeled arrows are used for functions, including transmission of energy across product interfaces, and for information flows. The concept of graphical mapping of useful and harmful actions or effects between the elements of a product structure and between such elements and users was originally published as part of a patent application filed by the TRIZ vendor Invention Machine Corporation (Devoino et al., 1997). The method was subsequently implemented in the Techoptimizer (now known as Goldfire) software and represented using five elements: component, supersystem, and product as types of blocks, and useful and harmful action as types of relationships. The TRIZ literature reports a limited number of applications. Among these, it is worth mentioning four studies to investigate the use of the method for the analysis and redesign of a car wheel, a window-cleaning process, a ducting system, and electronic products (Cascini & Rissone, 2004; Pinyayev, 2006; Aduka, 2010; Gadd, 2011).

2.3. Summary

The previous sections have presented a range of models for function analysis, focusing the discussion predominantly on the mainstream approach known as function structure. The FAD method was also presented as an approach to functional analysis that has been underresearched compared to the other methods. FAD differs from the functional models, which have received more attention (e.g., function tree and function structure), because it relies on product parts to model functions.

3. INTENDED BENEFITS OF THE FAD

Researching the FAD method, we identified seven intended benefits. These emerged from the analysis of FAD models in the literature (Cascini & Rissone, 2004, Pinyayev, 2006; Aduka, 2010; Gadd, 2011), as well as those developed by engineers in the collaborating company and the authors (Aurisicchio et al., 2012). The purpose of this article is not to demonstrate empirically that these benefits always hold true. In this article, we aim to introduce the intended benefits of the FAD method and justify the first six through a case study. Shortage of space precludes the exploration of the seventh benefit here, but it is addressed fully in Aurisicchio et al. (2012). The seven intended benefits of the FAD method are as follows:

1. The notation is simple and unobtrusive and in a sense intuitively obvious. This means that FAD modeling hardly needs to be explained because one can simply look at the diagram and understand it. This is an extremely important aspect to enable a wider diffusion of functional analysis.
2. The presence of the product structure makes the method easy to use.
3. The *mesh* representation with high internode connectivity allows a more complete description of functional relationships.
4. The layout of the diagram can be used to express additional meaning. The components of an assembly can be laid out following their actual positions. This is expected to be especially useful for design activities where space and position are relevant.
5. The diagram is useful to analyze an engineering system, capturing the rationale for why something is designed the way it is.
6. The diagram is a useful starting point for design improvement. Modeling functions together with product structure makes it suited to variant and adaptive design, unlike traditional approaches.
7. The representation of hierarchies of schematic function structures is feasible and practical. The structure and use of hierarchical FAD is demonstrated in Aurisicchio et al. (2012) using the hydraulic pump example reported by Eckert (Eckert et al., 2011; Eckert, 2013).

4. CASE STUDY OF FUNCTION ANALYSIS USING THE FAD

This section focuses on the main conceptual characteristics of the FAD method. It also justifies the first six intended benefits through a centrifugal water pump case study.

The FAD for the water pump is shown in Figure 1. The notation to represent the diagram is based on the *block* and *relation* elements to map, respectively, product structure, users, and other resources, and useful or harmful actions (Bracewell, Gourtovaia, et al., 2009). As can be seen in Figure 1, the model includes 11 blocks with dark background to represent the components of the pump (e.g., pump body and pump lid), 4 blocks to model the liquid flow in different points in time and location (e.g., inlet water flow), and 2 blocks to model components upstream and donwstream of the pump (e.g., electric motor and piping system). There are approximately 26 relations, of which 20 are useful (e.g., shaft drives impeller) and 6 are harmful (e.g., lip seal generates friction on shaft).

The diagram can be read starting from any of the block elements in the map. The FAD model in Figure 1 was created by reverse engineering, as were almost all functional models in the literature. Starting from the physical model, blocks of the components and water were laid out in the canvas and relations between them set.

The FAD is now compared to the function structure in order to relate its characteristics to the mainstream approach to function analysis. A function structure model for the centrifugal water pump can be seen in Figure 2, adapted from a model in the Design Repository at Oregon State University (Stone, 2012). Although the notations are apparently similar, closer inspection reveals fundamental differences. The FAD model is less abstract than the function structure model because it relies on product structure to model functions. Another element contributing to the FAD model being less abstract is the vocabulary employed (see Fig. 1). This is based on natural language and therefore easy to understand and use. It can be seen, for example, that actions are described using up to five words (e.g., pump body *generates adverse pressure gradient in* scroll water flow). By contrast, the function structure model uses the RFB vocabulary. Although such vocabulary is important to achieve the objective of creating models that can offer reusable knowledge for future projects, it can be argued that it requires a long learning curve before users can benefit from it.

Comparing the two models, it also appears that the function structure model captures a predominantly linear mesh of relations, while the FAD model is much richer and captures a mesh of interconnected actions.

Another important characteristic of the FAD is that the layout can have meaning. The elements of the diagram can be laid out in an arbitrary manner as well as according to a specific pattern that may, for example, follow the layout of the physical components of the system being analyzed. Figure 3 shows an example of the FAD for the water pump organized over the layout of the actual pump body.

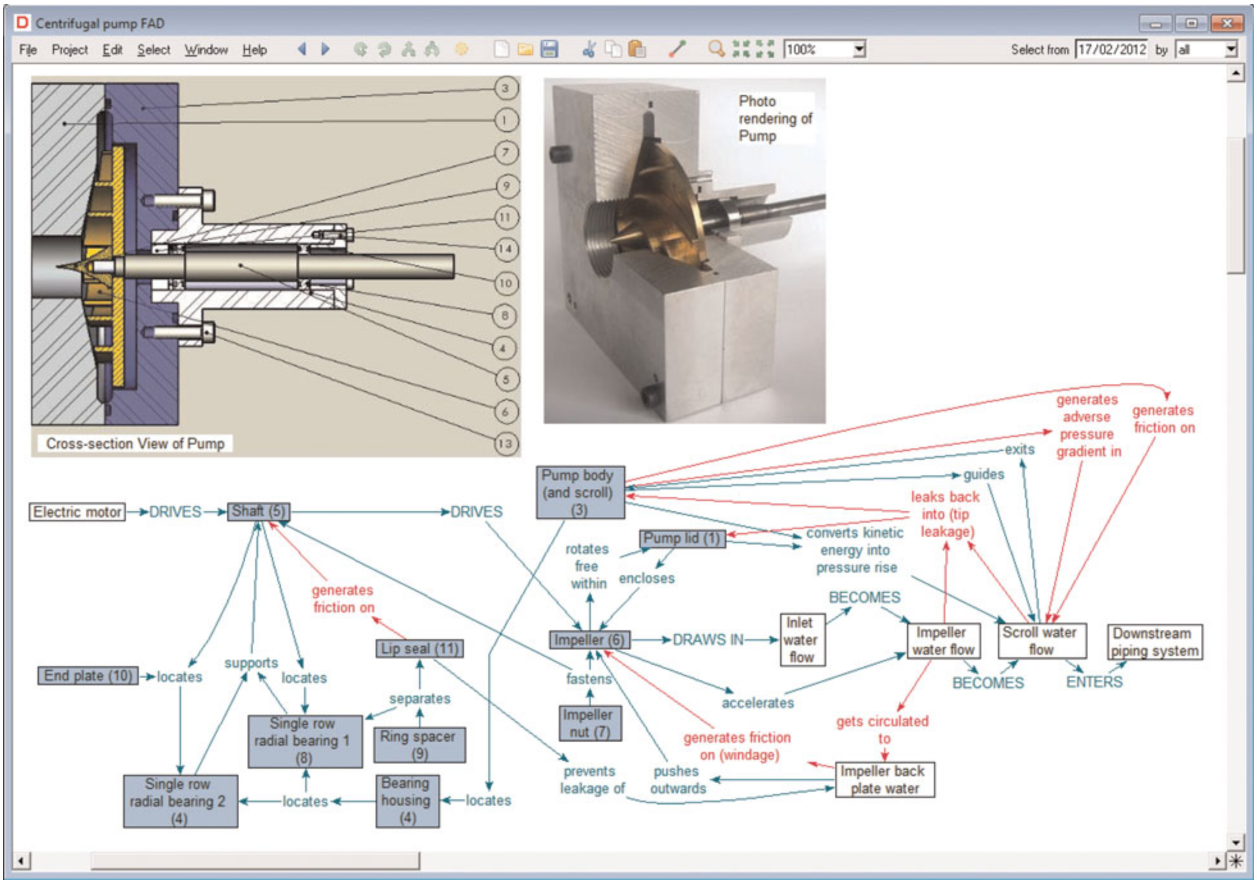

Fig. 1. A function analysis diagram (FAD) of a water pump. [A color version of this figure can be viewed online at http://journals.cambridge.org/aie]

The functional information captured in a FAD is not unique to this method. The second house of the four-house quality function deployment method (QFD2) captures the same relationships using a traceability matrix (see Fig. 4). The matrix is surely more compact than the FAD representation, but it is also less visual. FAD and QFD2 seem to provide, together, complementary views on the analysis of functional interactions.

The FAD model helps engineers in the analysis of new or existing systems by supporting the documentation and visualization of functional interactions. These interactions capture an aspect of rationale for why the pump is designed the way it is (Lee, 1997). To support this point, it is now worth reflecting on the fact that by reading the cross section of the solid model in Figure 1, an engineer has to infer the relationships between the pump components, whereas the diagram makes them explicit.

With the proportion of variant and adaptive design work together being significantly larger than that of original design work (Court, 1995), modeling function interactions with product structure seems a promising solution to support the work of engineers in industry. In this respect, the FAD method offers a more practical solution to understand functional interactions and solve potential undesired effects. As shown earlier, the diagram allows the characterization of a design problem by representing harmful functions. It, therefore,

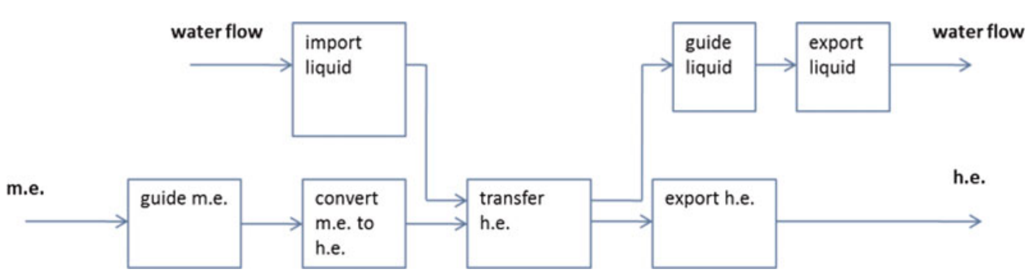

Fig. 2. A function structure of a water pump (m.e., mechanical energy; h.e., hydraulic energy). [A color version of this figure can be viewed online at http://journals.cambridge.org/aie]

https://doi.org/10.1017/S0890060413000255 Published online by Cambridge University Press

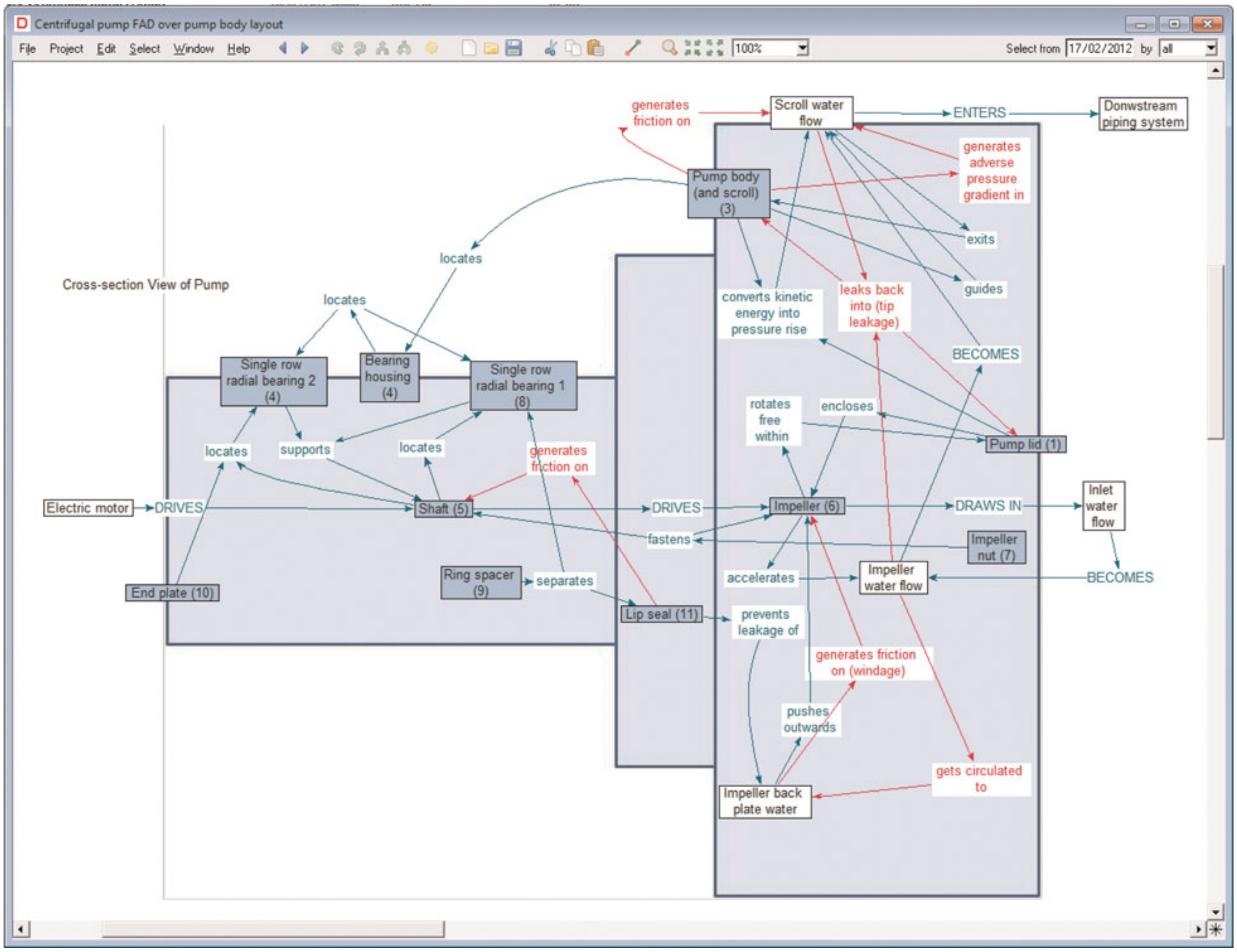

Fig. 3. A function analysis diagram (FAD) over a pump body layout. [A color version of this figure can be viewed online at http://journals.cambridge.org/aie]

	pump lid	pump body (and scroll)	impeller	impeller nut	shaft	bearing housing	lip seal	ring spacer	single row ball bearing 1	single row ball bearing 2	end plate
drive impeller					Δ						
draw in inlet water flow			Δ								
prevent leakage of impeller back plate water							Δ				
support shaft									Δ	Δ	
locate single row ball bearing 1					Δ	Δ					
locate single row ball bearing 2					Δ	Δ					Δ
separate lip seal and ball bearing 1								Δ			
locate bearing housing		Δ									
fasten impeller				Δ							
accelerate impeller water flow			Δ								
rise pressure of scroll water flow	Δ	Δ									
guide scroll water flow	Δ	Δ									

Fig. 4. Quality function deployment 2 for a water pump. The delta (Δ) means that a relationship exists between a row and a column.

https://doi.org/10.1017/S0890060413000255 Published online by Cambridge University Press

offers a starting point for engineering tasks requiring design improvement.

5. DISCUSSION

The results of this research have shown that modeling product functionality together with structure produces models that are less abstract and more intuitive than form-independent models. This is a critical precondition to develop a method that can be widely taken up by engineers in industry. A possible reason for the FAD model being more intuitive is that it better aligns with the natural way of working of engineers involving simultaneous thinking with function and structure. Another reason is that in FAD modeling functions are expressed through natural language, not the verbs and nouns of the RFB.

The results have also shown that the FAD model captures a richer set of functions than the function structure. This level of modeling appears more suitable to represent real-world engineering design problems and offers a more concrete support in the analysis of new and existing systems. The centrifugal water pump case study has also demonstrated that the FAD model captures in an easy-to-read format an aspect of the pump design rationale, and in this way it makes explicit knowledge, which would not normally be documented. This explanatory rationale can be considered complementary to that captured through issue-based information system structures, where the focus would be on documenting solution alternatives to achieve each of the pump functions (Bracewell, Wallace, et al., 2009). The combination of these two types of rationale seems to have the potential to support knowledge management strategies to create richer product information repositories.

By modeling not just useful actions but also harmful ones, the method offers a starting point to propose design improvements. This indicates that FAD is more suitable to support variant and adaptive design work than are form-independent models, which instead have been researched to aid original design by abstraction and analogy. Despite the fact that the application of the FAD method presented in this article does not provide an answer to the problem of supporting original design, practitioners have argued that the method can be useful also in this context (Adunka, 2010).

Using the reasoning scheme in Brown and Blessing (2005), the relations mapped in the water pump FAD in Figure 1 can be classed at the function level. This is the accepted meaning of function in the collaborating company, and it does not have to coexist with others. This meaning is compatible with that used in the application of other design methods (e.g., QFD and failure mode effect analysis) and controlled by the generic design practice (Rolls-Royce, personal communication, 2009), a structured process for method-based design of complex power systems. In FAD modeling by engineers in our partner company, there has not been a need to link the method to a meaning of function because this is typically defined by the product designed (i.e., power systems for use in the air, on land, and at sea) and the task (i.e., variant and adaptive design).

It is noteworthy that the meaning of function used in the water pump FAD is also that adopted by proponents of the function structure and the RFB. There are functional expressions in the FAD model in Figure 1 and the function structure in Figure 2 that correspond and differ only because the terms used to describe them are at different levels of abstraction and subject or not to specific constraints. Therefore, what determines the difference between the models is the representation and the modeling choices related to the task.

Although more work is needed to demonstrate its usefulness, research by the authors indicates that the FAD method can be used also with functional information at the action and goal level of the reasoning scheme in Brown and Blessing (2005). For example, in Aurisicchio et al. (2011), a lemon squeezer FAD is shown in which the relations mapped can be classed at both the function and the action levels of the reasoning scheme. Hence, multiple meanings of function can be accepted with FAD modeling and at times also used simultaneously. In the authors' opinion, no matter whether one or multiple concepts of function are used in a FAD model, this is unlikely to generate ambiguity because the meaning is determined by the product designed and the task (Vermaas, 2013).

Overall, it seems that in the current functional modeling tool set, there is a place and a need for a pragmatic method such as FAD (Eckert, 2013). The results of this research do not suggest that FAD is better than the other functional models. Rather, this research indicates that the goodness of a model is task specific (Goel, 2013; Vermaas, 2013). FAD can coexist with form-independent models and enrich the tool set available to engineers for function analysis.

We stated earlier in this article that FAD has received little attention compared to the function structure and other functional models. It is now worth asking why research has followed this path. We argue that the answer is to be found in a combination of issues, including the goal of the research community, the origin of the modeling formalism employed, and the methodological approach adopted. The design community behind the function structure has focused predominantly on tackling the challenging problem of supporting original design. Although much of this work has been practical, adopting a reverse engineering approach, researchers have focused on simple problems and have continued using a model that is traceable to the work of system theorists. The artificial intelligence in design community has long moved in the direction of modeling product structure together with behavior and function but with a focus on formal, mostly symbolic, representations that can support computational reasoning. In addition, the adoption of the "walk before you run principle" has meant that they have largely modeled relatively simple rather than industrial-strength problems in areas like adaptive design and model-based diagnosis. For FAD to emerge, a more practically focused community was needed, which has materialized around the TRIZ methodology, its bottom-up methodological approach, and industrial problems.

5.1. Limitations and further work

The results of this study are based on research in collaboration with engineers in industry and are illustrated using a case study developed by the authors. More work is needed to evaluate the method and learn from engineering applications. Further research will focus on: the development of the FAD syntax and vocabulary; the extension of the method to model functions across different product states (time dimension in function analysis); and the extension of the method to represent possible product configurations (contingency dimension in function analysis).

5.2. Practical and theoretical implications

FAD is a method that can be used now to map and understand function interactions in engineering systems. As much as for other modeling tools (e.g., CAD), capabilities vary significantly depending on the chosen application. FAD models can be created through tools like DRed (owned and controlled by Rolls-Royce plc), designVUE (open source tool freely downloadable from the Design Engineering group at Imperial College), and most concept mapping tools.

This paper contributes to engineering design research at two levels. First, it brings to the attention of academics and practitioners a method for function analysis used in industry that has been underresearched compared to mainstream methods for function analysis. Second, it advances current understanding of the method by presenting its characteristics and justifying its advantages and disadvantages through an engineering case study.

6. CONCLUSIONS

This research was undertaken with the aim of investigating computer-based modeling of functional interactions in engineered systems. The results of the literature review have shown that the support offered by current methods is limited. Mainstream methods have predominantly attempted to model product functionality independently from structure. It was argued that form-independent functional representations are compatible with engineers' natural way of working, which consists of shifting between function and form-based reasoning. The FAD method brought to the attention of the authors by engineers in the collaborating company was found to have several promising characteristics. The results of the research have shown that the FAD method can be used to create rich and easy-to-use functional models and that it can coexist with current functional modeling tools.

ACKNOWLEDGMENTS

The authors acknowledge the support of the Technology Strategy Board and Rolls-Royce plc through the Strategic Investment in Low-Carbon Engine Technology project.

REFERENCES

Adunka, R. (2010). Function analysis for electronic products. *Proc. TRIZ Future Conf.*, pp. 165–171, Bergamo, Italy, November 3–5.

Araujo, C.S., Benedetto-Neto, H., Campello, A.C., Segre, F.M., & Wright, I.C. (1996). The utilization of product development methods: a survey of UK industry. *Journal of Engineering Design 7(3)*, 265–277.

Aurisicchio, M., Bracewell, R., & Armstrong, G. (2012). Functional modeling through the function analysis diagram, DETC2012-70944. *Proc. ASME 2012 Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf., IDETC/CIE 2012*, Chicago, August 12–15.

Aurisicchio, M., Eng, N.L., Ortiz Nicolas, J.C., Childs, P.R.N., & Bracewell, R.H. (2011). On the functions of products. *Proc. 18th Int. Conf. Engineering Design (ICED 11)*, pp. 443–455. Lyngby/Copenhagen: Design Society.

Bertalanffy, L.V. (1969). *General System Theory: Foundations, Developments, Applications* (rev. ed.) New York: Braziller.

Bhatta, S., Goel, A., & Prabhakar, S. (1996) Innovation in analogical design: a model-based approach. *Proc. AI in Design*. Dordrecht: Kluwer Academic.

Bonaccorsi, A., Apreda, R., & Fantoni, G. (2009). A theory of the constituent elements of functions. *ICED 09*, pp. 179–190, Stanford, CA, August 24–27, 2009.

Bracewell, R., Gourtovaia, M., Moss, M., Knott, D., & Wallace, K. (2009). DRed 2.0: a method and tool for capture and communication of design knowledge deliberated in the creation of technical products. *ICED 09*, pp. 223–234, Stanford, CA, August 24–27.

Bracewell, R., & Sharpe, J.E.E. (1996). Functional descriptions used in computer support for qualitative scheme generation—Schemebuilder. *Artificial Intelligence in Engineering Design, Analysis and Manufacturing 10(4)*, 333–346.

Bracewell, R., Wallace, K., Moss, M., & Knott, D. (2009). Capturing design rationale. *Computer-Aided Design 41(3)*, 173–186.

Brown, D.C., & Blessing, L. (2005). The relationship between function and affordance. *ASME 2005 Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf. IDETC/CIE 2005*, pp. 155–160. Long Beach, CA, September 24–28.

Bryant, C.R., McAdams, D.A., Stone, R.B., Kurtoglu, T., & Campbell, M.I. (2005). Automation of a computational technique for concept generation. *ASME IDETC/CIE 2005*, pp. 267–276, Long Beach, CA, September 24–28.

Cascini, G., & Rissone, P. (2004). Plastic design: integrating TRIZ creativity and semantic knowledge portals. *Journal of Engineering Design 15(4)*, 405–424.

Collins, J., Hagan, B., & Bratt, H. (1976). The failure–experience matrix—a useful design tool. *Transactions of the ASME, Series B, Journal of Engineering in Industry 98*, 1074–1079.

Court, A.W. (1995). *Modelling and classification of information for engineering designers*. PhD Thesis. University of Bath.

Crilly, N. (2010). The roles that artefacts play: technical, social and aesthetic functions. *Design Studies 31(4)*, 311–344.

Devoino, I.G., Koshevoy, O.E., Litvin, S.S., & Tsourikov, V. (1997). *Computer-based system for imagining and analysing an engineering object system and indicating values of specific design changes.* US Patent 6056428.

Dori, D. (2002). *Object–Process Methodology—A Holistic Systems Paradigm*. New York: Springer–Verlag.

Eckert, C. (2013). That which is not form: the practical challenges of using functional concepts in design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 27(3)*, 217–232 [this issue].

Eckert, C., Alink, T., Ruckpaul, A., & Albers, A. (2011). Different notions of function: results from an experiment on the analysis of an existing product. *Journal of Engineering Design 22(11–12)*, 811–837.

Fantoni, G., Apreda, R., & Bonaccorsi, A. (2009). Functional vector space. *Proc. ICED 09*, pp. 311–322, Stanford, CA, August 24–27.

Gadd, K. (2011). *TRIZ for Engineers—Enabling Inventive Problem Solving*. New York: Wiley.

Goel, A.K. (2013). A 30-year case study and 15 principles: implications of an artificial intelligence methodology for functional modeling. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 27(3)*, 203–215 [this issue].

Hirtz, J., Stone, R., McAdams, D., Szykman, S., & Wood, K. (2002). A functional basis for engineering design: reconciling and evolving previous efforts. *Research in Engineering Design 13*, 65–82.

Hubka, V., & Eder, W.E. (1984). *Theory of Technical Systems*. Berlin: Springer–Verlag.
Hundal, M. (1990). A systematic method for developing function structures, solutions and concept variants. *Mechanism and Machine Theory 25(3)*, 243–256.
Iwasaki, Y., Fikes, R., Vescovi, M., & Chandrasekaran, B. (1993). How things are intended to work: capturing functional knowledge in device design. *Proc. Int. Joint Conf. AI*, pp. 1516–1522. Menlo Park, CA: AAAI Press.
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.
Lee, J. (1997). Design rationale systems: understanding the issues. *IEEE Expert: Intelligent Systems and Their Applications 12(3)*, 78–85.
Lefever, D., & Wood, K. (1996). Design for assembly techniques in reverse engineering and redesign. *Proc. ASME Design Theory and Methodology Conf.*, Paper No. DTM-1507, Irvine, CA.
Little, A., Wood, K., & McAdams, D. (1997). Functional analysis: a fundamental empirical study for reverse engineering, benchmarking and redesign. *Proc. ASME Design Engineering Technical Conf.*, Paper No. DTM-3879, Sacramento, CA.
Lopez-Mesa, B., & Bylund, N. (2011). A study of the use of concept selection methods from inside a company. *Research in Engineering Design 22(1)*, 7–27.
Miles, L. (1972). *Techniques for Value Analysis and Engineering*. New York: McGraw–Hill.
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*. London: Springer–Verlag.
Pinyayev, A. (2006, December). Functional clues. *TRIZ Journal*.
Simon, H.A. (1996). *The Sciences of the Artificial*. Cambridge, MA: MIT Press.
Stone, R. (2012). *Design Repository*. Oregon State University, Design Engineering Lab. Accessed February 19, 2012, at http://designengineeringlab.org/delabsite/repository.html
Stone, R., & Wood, K. (2000). Development of a functional basis for design. *Journal of Mechanical Design 122(4)*, 359–370.
Suh, N.P. (2001). *Axiomatic Design: Advances and Applications*. New York: Oxford University Press.
Szykman, S., Racz, J.W., & Sriram, R.D. (1999). The representation of function in computer-based design. *Proc. ASME Design Engineering Technical Conf.*, Paper No. DTM-8742, Las Vegas, NV, September 12–15.
Ullman, D. (1992). *The Mechanical Design Process*. New York: McGraw–Hill.
Ulrich, K.T., & Eppinger, S. (1995). *Product Design and Development*. New York: McGraw–Hill.
Umeda, Y., & Tomiyama, T. (1997). Functional reasoning in design. *IEEE Intelligent Systems 12(2)*, 42–48.
Value Analysis Incorporated. (1993). *Value Analysis, Value Engineering, and Value Management*. Clifton Park, NY: Value Analysis Incorporated.
Vermaas, P.E. (2013). The coexistence of engineering meanings of function: four responses and their methodological implications. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 27(3)*, 191–202 [this issue].
Whybrew, K., Shaw, A.I., Aitchison, D.R., & Raine, J. K. (2001). Use of design tools and methodologies for rapid product development in the New Zealand manufacturing industry. *Proc. 13th Int. Conf. Engineering Design, Design Applications*, pp. 27–34, Glasgow.
Yourdon, E. (1989), *Modern Structured Analysis*. Upper Saddle River, NJ: Prentice–Hall.

Marco Aurisicchio is a University Lecturer in the Mechanical Engineering Department at Imperial College and a Senior Tutor in the Innovation Design Engineering Programme at the Royal College of Art. Prior to joining Imperial College, Marco was a Research Associate at the University of Cambridge, where he also undertook his PhD in engineering design. His research interests include the design process of complex systems and knowledge management in product development. Dr. Aurisicchio has worked with Rolls-Royce plc to improve its design processes for over 10 years. His work with the company has contributed to the development of the DRed, a software tool for design knowledge capture.

Rob Bracewell is a Senior Research Associate at University of Cambridge and a Fellow of Churchill College. For over 20 years in Cambridge and previously Lancaster Engineering Design Centres, he has researched and implemented software tools to aid designers. His goal is to provide teams addressing complex problems with effective, properly integrated methodical computer support for the whole design process, not just the later stages where traditional CAD excels. For the past decade he has continuously collaborated with Rolls-Royce plc on the development of DRed, which has steadily become an important part of the company's engineering tool set.

Gareth Armstrong is a Design Engineer at Rolls-Royce plc, Derby. In the 10 years since he joined the company, he has been involved in all aspects of the design process on a range of gas turbine projects in the United Kingdom, Germany, and the United States. He is currently working on the Design Methods team, where his responsibilities include developing and promoting improved processes, methods, and tools to support all aspects of the company's engineering activities.