Artificial Intelligence for Engineering Design, Analysis and Manufacturing (2017), *31*, 401–412.

doi:10.1017/S0890060417000476

Function in engineering: Benchmarking representations and models

JOSHUA D. SUMMERS,[1] CLAUDIA ECKERT,[2] AND ASHOK K. GOEL[3]

[1]Department of Mechanical Engineering, Clemson University, Clemson, South Carolina, USA
[2]Department of Design and Innovation, Open University, Milton Keynes, United Kingdom
[3]School of Interactive Computing, Georgia Institute of Technology, Atlanta, Georgia, USA

(RECEIVED November 8, 2016; ACCEPTED May 26, 2017)

Abstract

This paper presents the requirements and needs for establishing a benchmarking protocol that considers representation characteristics, supported cognitive criteria, and enabled reasoning activities for the systematic comparison of function modeling representations. Problem types are defined as reverse engineering, familiar products, novel products, and single-component systems. As different modeling approaches share elements, a comparison of modeling approaches on multiple levels was also undertaken. It is recommended that researchers and developers of function modeling representations collaborate to define a canonically acceptable set of benchmark tests and evaluations so that clear benefits and weaknesses for the disparate collection of approaches can be compared. This paper is written as a call to action for the research community to begin establishing a benchmarking standard protocol for function modeling comparison purposes. This protocol should be refined with input from developers of the competing approaches in an academically open environment. At the same time, the benchmarking criteria identified should also serve as a guide for validating a modeling approach or analyzing its failure.

Keywords: Benchmarking; Engineering; Function; Models; Representation Characteristics

1. INTRODUCTION

Reasoning about function of products is critical in product development, which has led to many approaches to functional modeling being advocated to support systematic products development. The selection of the approach deemed most appropriate for a particular reasoning need remains problematic, however, given the lack of clear guidance for informing this decision (Otto & Wood, 2001; Ulrich & Eppinger, 2008; Ullman, 2010; Pahl et al., 2013). Therefore, this paper provides a justification and a proposed research direction for establishing a common benchmarking scheme for function representations that are developed and deployed throughout academia and practice with the ultimate goal of providing industry with practically usable functional modeling tools and concepts and a clear rationale for selecting a particular one. Despite decades of research into functional descriptions, research suggests industry has yet to incorporate functional modeling in practice in a systematic way, while still proclaiming a need to express product information beyond form (Eckert, 2013; Tomiyama et al., 2013; Arlitt et al., 2016). It should be noted that there are other representations that are similar, but are not directly associated with function modeling, that are used in industry such as p-diagrams (Telenko & Seepersad, 2010; Campean et al., 2013), block diagramming (Sturges et al., 1996; Braha & Maimon, 1998), or IDEF0 (Nagel et al., 2009; Buede & Miller, 2016). Another possible reason for this resistance is the lack of a canonical definition of function with each approach grounded in different conceptualizations, or the possibility of multiple distinct concepts inherent within a shared terminology.

Researchers and practitioners have proposed many different views of function in engineering design (Deng, 2002; Crilly, 2010; Srinivasan et al., 2012; Eckert, 2013; Goel, 2013; Vermaas, 2013), with three recent approaches included in this Special Issue on function being the dimensional analysis conceptual modeling framework (Hossein et al., 2017), critical chain models (Agyemang et al., 2017), and system state flow diagrams (Yildirim et al., 2017). These concepts in turn have been used to inform the creation of many approaches for modeling information about a product's functionality. For example, several design textbooks emphasize the use of function-flow networks to capture the sequence

Reprint requests to: Joshua D. Summers, 203 Engineering Innovation Building, Fluor Daniel, Department of Mechanical Engineering, Clemson University, Clemson, SC 29634-0921, USA. E-mail: jsummer@clemson.edu

and dependencies for the desired functionality of a product (Otto & Wood, 2001; Ulrich & Eppinger, 2008; Ullman, 2010; Pahl et al., 2013).

Rather than develop a single, unified definition of function, we assert that each approach has its own strengths and weaknesses. Although each approach is useful and particularly well suited for different reasoning applications and domains, their transference across different domains remains a difficult proposition. Therefore, we propose a different approach to function research through the creation of a set of comparative benchmarks that can be explored with the different modeling approaches. The community may in turn use these proposed benchmarks to discern which approaches are more useful for different needs, and perhaps to discover which elements of the representations and vocabularies are most conducive for different elements of functional thinking.

The information captured within function models can be used to facilitate many different engineering activities across the entire product lifecycle, such as synthesis, analysis, exploration, visualization, explanation, and fault detection (Gero & Kannengiesser, 2002; Goel & Bhatta, 2004; Kurtoglu & Tumer, 2008). While modeling approaches might be defined clearly, they do not always come with clear guidance on how to represent specific models. Unfortunately, these modeling approaches and representations are perceived as easy neither to use nor to learn, with potential users remaining poorly informed with respect to what these representations can provide. Therefore, both of these assumptions represent an educational challenge for the community with the notion of function even considered as unclear (Eckert et al., 2011). Thus, these assumptions must be addressed with each representation and modeling approach proposed, but with common frames of reference, perhaps supported through a standard benchmarking protocol that would define common problems and common issues against which methods can be challenged.

2. THE NEED FOR BENCHMARKING

We next turn to other canonical benchmark systems that have proven useful in the cross-comparison of algorithms and methods. Despite the subject of repeated analysis in the research community, benchmarking has rarely been applied to the methods and tools developed in the engineering design community. One of the few examples of an effort to directly compare different function representations focused on additive manufacturing processes (Summers & Rosen, 2013). For example, software engineering may entail the use of benchmarking to compare algorithms (Dolan & Moré, 2002), specifically with regard to the traveling salesman problem, which involves a series of benchmark problems used for such comparisons (Peterson, 1990). Likewise several accepted optimization benchmark problems have been used to evaluate performance of new algorithms (Brest et al., 2006). Similarly, standard benchmark tests have been used in the automotive industry to consider different control strategies (Rajamani, 2012).

Each of these different benchmark sets have been constructed to test new algorithms, either optimization or controls. With developing a benchmark set of problems for comparing function representations, the algorithm, or reasoning dimension, must be considered, as should the representation and the modeling of the functions, thus defining a critical distinction between the traditional approaches of benchmarking and the approach proposed here.

No protocol for benchmarking functional modeling currently exists, nor has there been any systematic comparison of the expressive power of various models been, despite the individual comparisons available in the literature of such relationships. A huge variability between individuals and in particular between different modeling approaches exists, as indicated in the experiment on functional descriptions, where different engineers were provided with a product and asked to generate a functional description (Eckert et al., 2012). Consequently, a systematic analysis is needed to determine how different modeling approaches compare in representational expressive power, reasoning inferencing capacity, and modeling ease of use.

3. IS FUNCTION RESEARCH SUFFICIENTLY MATURE?

Before defining a series of benchmark test cases for use in the cross-evaluation of competing function representations, we must first determine if the research field is sufficiently established to warrant such an effort. This is critical for determining both the sufficient need in a plethora of competing approaches and a sufficient population size of researchers willing to use these benchmarks as comparative tools. To this end, we first consider the field's evolution over the past five decades (Table 1). The evolution has been characterized by incremental shifts and the creation of function vocabularies (from normative function descriptions, Pahl et al., 2013; to controlled vocabularies, Hirtz, Stone, McAdams, et al., 2002; to physics-defined vocabularies, Sen et al., 2013*c*). It has also involved the creation of conceptually divergent approaches for modeling interface-centered (Wang et al., 2009), component-centered (Fenves et al., 2008), and user-centered functionalities (Gaffney et al., 2007). Functional modeling currently receives a renewed interest as the greater integration of mechanical systems with electric systems and software required trade-off across systems on a functional. For example, the integrated function modeling framework (Gericke & Eisenbart, 2017) in this Special Issue bridges this gap by combining multiple views on functional modeling.

There are many representations available, such as the structure–behavior–function (Bhatta & Goel, 1997), the function–behavior–structure (Qian & Gero, 1996), the functional basis (Hirtz et al., 2002), the function–behavior–state (Umeda et al., 1996), affordance-based design (Maier, Srinivasan, et al., 2007), the contact and channel model (Albers et al., 2008), and the general function lists used in such design tools

Table 1. *Recent decades of engineering function research*

Decade	Example References
1960s	Eastman, 1969; Pahl et al., 2013
1970s	Collins, Hagan, & Bratt, 1976; Freeman & Newell, 1971; Rodenacker, 1971
1980s	Andreasen & Hein, 1987; Hubka & Eder, 1988; Ullman, Dietterich, & Stauffer, 1988; Sembugamoorthy, Chandrasekaran, Sembugamoorthy, & Chandrasekaran, 1986
1990s	Bracewell & Sharpe, 1996; Goel, 1997; Kirschman & Fadel, 1998; Qian & Gero, 1996; Sasajima, Kitamura, Ikeda, & Mizoguchi, 1995; Umeda, Ishii, Yoshioka, Shimomura, & Tomiyama, 1996; Vescovi, Iwasaki, Fikes, & Chandrasekaran, 1993
2000s	Albers, Thau, & Alink, 2008; Chandrasekaran, 2005; Erden et al., 2008; Gero & Kannengiesser, 2002; Hirtz et al., 2002
2010s	Linz, 2011; Schultz et al., 2010; Sen, Summers, & Mocko, 2011; Srinivasan et al., 2012; Yang, Patil, & Dutta, 2010

as morphological charts (Smith et al., 2012), the house of quality (Olewnik & Lewis, 2005), and axiomatic design (Suh, 1999). Developed by many researchers worldwide, each of these function representations and modeling approaches is characterized with a different intent, history, and context behind the representation (Erden et al., 2008). It is because of the many different roles and uses within engineering design that these models have evolved disparately. Unfortunately, many of these approaches are limited by the "inventors" problem within design research, in which researchers will push the creation of a solution to a problem and a design need without actually designing the tool or method based on the intrinsic properties. Thus, many of the representations, while serving different specific purposes such as machinery and manufacturing systems with an emphasis on flows (Pahl et al., 2013), might support other activities addressed by competing representations. This discrepancy suggests the need for developing a systematic comparison system.

Most of the modeling approaches were developed in response to a problem that would have been difficult to approach using methods with which the authors were familiar, and that work particularly well on certain problems. For example, although the functional models based on flow of energy, matter, and information work extremely well for production machinery (Pahl et al., 2013), in which many examples are used to inform the model, they are ineffective in elucidating the functions a single complex component (e.g., the turbine blade of a jet engine). Further, although most of the functional modeling approaches in the literature use examples to illustrate their points, they neither discuss the scope of their approach nor reflect about the applicability to other classes of problems.

4. LEVELS OF COMPARISON

A thorough analysis of this problem must be predicated on a short discussion of the different levels at which function modeling, representation, and reasoning can be compared. Such comparisons may range from the fundamental core idea through to the use of models (Fig. 1). The most theoretical level of comparisons involves studying differences of "purpose," "transformation," or "intent" (Rosenman & Gero, 1998), which have been reduced to formal frameworks such as the function–behavior–structure (Gero & Kannengiesser, 2004). Such frameworks have been used to create defined representations for instantiating models, which are then coupled with reasoning activities, such as "model building" and "model using" for drawing inferences (Cebrian-Tarrason et al., 2008). Tools such as FunctionCAD (Nagel et al., 2009), 2nd-CAD (Vargas-Hernandez & Shah, 2004), function-behavior-state modeler (Umeda et al., 1996), design repository (Bohm et al., 2005), and ConMod (Sen et al., 2013*b*) have been used to support such model-building

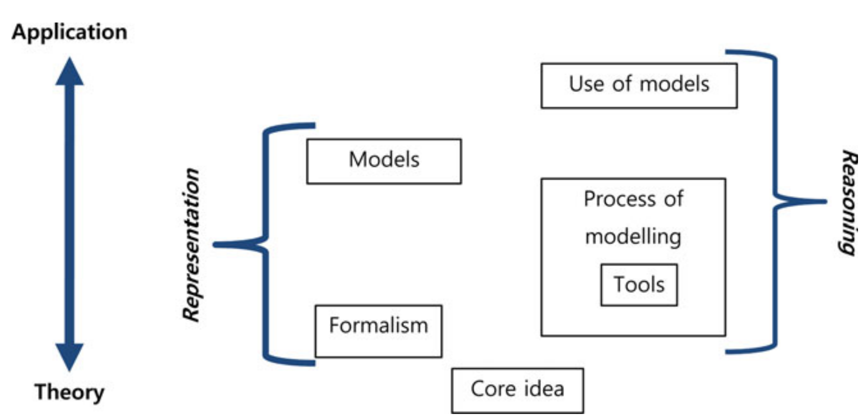

Fig. 1. Level of comparison and consideration during function benchmarking.

https://doi.org/10.1017/S0890060417000476 Published online by Cambridge University Press

activities. The models created from these tools have then been used for different inferences, such as failure modes (Stone et al., 2005), predicting assembly time (Mohinder et al., 2017), and predicting assembly time (Mathieson et al., 2011; Gill & Summers, 2016). Thus, we must consider the primary factors of representation and reasoning, and their interaction with the modeler and user.

Different functional modeling approaches have different core ideas at the core of the approach. For example, some analyze function is terms of the flow (Pahl et al., 2013), whereas other approaches involve elucidating the relationship of function, structure, and behavior (Gero & Kannengiesser, 2004). Underpinning the core idea of any of these approaches are various notions of functions (Vermaas, 2013).

The different modeling approaches usually are supplied with modeling formalisms, which are the parameters in which a functional model is described. For example, both verb–noun pairs (Hirtz, Stone, & McAdams, 2002; Nagel, Stone, et al., 2008; Nagel et al., 2009) and either sentences or single words have been used to describe a given function (Deng, 2002). Conversely, contact and channel have been used to prescribe the elements contained within the description of a function. One such formalism involves the description of at least two working surface pairs where the function is enacted (Albers et al., 2008).

The expressive power of these functional models is greatly affected by such formalisms that are used in their construction, a variation of great importance when building functional models of complex products, where the effort increases significantly with the complexity of a model. Some modeling approaches are characterized by a hierarchical decomposition of functions while others are not (Erden et al., 2008). The models also vary in the degree of abstraction required; for example, contact and channel and function modeling requires a concrete embodiment of the modeling link functions, whereas functional descriptions are abstract in other models. This variation in turn affects the hierarchical decomposition as the lower level description might depend on the chosen embodiment.

The other dimension of comparison is the reasoning with functions both in building models and in using models once they have been generated, which is of importance when the models are used by someone other than their author. In such a case, the familiarity of the modeling approach and the intuitiveness of the representation then becomes a major issue when inferring information. Several studies have been undertaken to study the authorship and consistency and interpretability of models (Kurfman et al., 2003; Caldwell, Ramachandran, et al., 2012; Caldwell, Thomas, et al., 2012), as well as elucidating the correctness of model construction (Nagel et al., 2015). Alternatively, some approaches have been proposed that automatically reason on function models from database collections (Lucero et al., 2014; Patel, Andrews, et al., 2016; Sridhar et al., 2016). Finally, some approaches entail the support of first principle based physics reasoning (Goel et al., 2009; Sen et al., 2011*b*, 2013*a*). Thus, it is possible that reasoning might be supported through human use and interpretation or through automated reasoning to infer information.

Some functional modeling approaches, such as functional basis (Hirtz et al., 2002), are supported by dedicated modeling tools (Vargas-Hernandez & Shah, 2004; Bryant et al., 2006; Nagel et al., 2009), which allow the user to build models more comfortably and therefore also support building larger models. The usability of the tools affects the potential success of the modeling approach independent of the other elements.

5. COMPARISON ACROSS THE CRITERIA

The framework in Figure 1 proposes different criteria on which modeling approaches can be compared. Here we detail our criteria for comparing these criteria by proposing characteristics along which the comparison can be made.

5.1. Representation characteristics

When comparing function modeling approaches, the typical approach initially involves comparing the representations at the formalism level, including vocabulary and grammar. A representation is the formalism through which a model is constructed, which means that a model is instantiated through a representation. Multiple models of the same "real-world" target may be created through the same representation with each distinct model clearly mapped to a single real-world target. Ambiguity is introduced when one model serves as a surrogate for multiple targets (Shah & Mantyla, 1995), which are not clearly specified.

Similar comparison criteria, derived from research in artificial intelligence, include representational adequacy, inferential adequacy, inferential efficiency, and acquisitional efficiency (Winston, 2005). Another approach to compare representations examines the vocabulary, structure, expression, purpose, and abstraction (Summers & Shah, 2004). Expanding upon that research, we propose that the representation comparison should include, but not be limited to the following:

- *scope:* the domain for which the function modeling approach is intended (Nagel, Vucovich, et al., 2008);
- *flexibility:* the ability to modify and adapt the representation to address new problems (Regli et al., 2000);
- *indexing:* support access to the right (or useful) knowledge when needed (Goel & Bhatta, 2004);
- *consistency:* enforce physics and other consistencies (Sen et al., 2011*b*);
- *translationabilty:* tied to other engineering models (Nebel, 2000);
- *behavior:* ability of the representation to simulation behavior (Qian & Gero, 1996); and
- *scalability:* support both simple and complex problem types (Chiang et al., 2001).

5.2. Modeling characteristics

In addition to these representational issues, the interaction of the designer during the model construction is of concern when comparing the functional modeling approaches. For instance, is the modeling computationally supported, restricted to human effort, or is a mixed initiative approach supported (Sen et al., 2013*a*). In addition, the support of various construction approaches within the model, such as forward chaining (moving from input to output), backward chaining (moving from output to input), nucleation, environment to system (outside to inside), or system to environment (inside to outside) are also important considerations (Sen & Summers, 2014). A final characteristic relates to the support of decomposition and recomposition across multiple hierarchical levels and abstractions within the respective modeling approach (Pahl et al., 2013).

5.3. Cognitive criteria characteristics

The concept of cognitive criteria has been developed in human–computer interaction to help software designers to think through the usability of the artifacts they were creating, such as programming languages or user interfaces (Green, 1991). Although many developers and software engineers have experience in developing well-design information artifacts, they have no way of articulating why these approaches are appropriate for meeting user needs. Functional modeling approaches are considered information artifacts, similar to how programming languages and cognitive criteria offer a vocabulary for discussing usability issues, which is informed by cognitive science (Blackwell et al., 2001). The framework is deliberately broad to avoid being overwhelmed in the details of an implementation and thereby losing the sought-after conceptual improvements. However, this task-specific approach, which addresses processes and activities rather than merely the final product, means that it can be used to evaluate and not simply compare *functional modeling approaches.* The cognitive criteria are orthogonal in supporting reasoning trade-offs and for analyzing the space of possible solutions in a coherent manner, and where possible observing the effect of combinations of criteria. Table 2 presents a selection of the cognitive criteria with their questions for programming and a possible interpretation of these questions for functional modeling, which would require refinement prior to a benchmarking exercise (Green & Petre, 1996).

The computing cognitive criteria have a dimension of the progressive evaluation, which is the method for obtaining feedback on the modeling through the process and that appears far more meaningful for a programming language, which can be deployed in many different ways. Criteria such as diffuseness, which address the number of symbols or graphic entities required to express a meaning, and hard mental operations, which questions the need for annotations, also specifically address the notion. Both greatly depend upon a particular implementation version, as few standards concerning functional modeling have yet to emerge.

For each of the dimensions a scale of subcategories can be developed such as the abstraction gradient, which is decomposed into abstraction-hating, abstraction-tolerant, and abstraction-hungry (Green & Petre, 1996). For example, while abstraction-hungry programming languages may be considered difficult, abstraction can reduce error proneness and increase viscosity.

5.4. Reasoning characteristics

Reasoning is the comparison dimension that motivates the need for a common, standard benchmark for evaluating function modeling approaches. It is for different classes of reasoning that each function model is constructed. These rea-

Table 2. *Key cognitive dimensions (criteria) based on Green and Petre (1996)*

Dimensions	Question for Programming Languages	Question of Functional Modeling
Abstraction gradient	What are the minimum and maximum levels of abstraction? Can fragments be encapsulated?	What are the minimum and maximum levels of abstraction? Can a partial model be created?
Closeness of mapping	What "programming games" must be learned?	What modeling conventions must be learned? How intuitive is the resulting model?
Error proneness	Does the design of the notation induce "careless mistakes"?	Does the design of the notation induce "careless mistakes"?
Hidden dependencies	Is every dependency overtly indicated in both directions? Is the indication perceptual or only symbolic?	Is every dependency overtly indicated in both directions? Is the indication perceptual or only symbolic?
Premature commitment	Must programmers make decisions before they have the information they need?	Does the model require decisions prior to availability of all necessary information?
Secondary notation	Can programmers use layout, color, or other cues to convey extra meaning, above and beyond the "official" semantics of the language?	Can the models be annotated or linked to other product representations?
Viscosity	How much effort is required to perform a single change?	How much effort is required to perform a single change? What is the difficulty of adapting the model from a model of a similar product?

https://doi.org/10.1017/S0890060417000476 Published online by Cambridge University Press

soning activities entail failure detection (Kurtoglu & Tumer, 2008), reverse engineering and product understanding (Hirtz et al., 2002), design decision justification (Gero, 1996), design verification and validation (Wiltgen & Goel, 2016), or concept definition and exploration (Pahl et al., 2013). Some types of reasoning that can be evaluated with respect to support include the following:

- *Interpretability:* How consistent and precise is the interpretation of the function models across different individuals, domain, and expertise (Caldwell, Thomas, et al., 2012)?
- *Physics maintenance:* Can questions about conservation of energy or material, irreversibility, or other physics queries be answered (Sen et al., 2011*a*)?
- *Analogical mapping:* Does the representation support analogical mapping and alignment (Qian & Gero, 1996)?
- *Pattern learning:* Does the representation support the learning of abstractions required for analogical transfer (Bhatta & Goel, 1997)?
- *State transformations:* Does the representation support answering questions about different states (Deng, 2002)?
- *Change propagation:* Does the representation support discovery about the effects of perturbations in the system (Kurtoglu & Tumer, 2008)?

These reasoning criteria might relate to the cognitive criteria. For example, interpretability, analogical mapping, and change propagation might relate to closeness of mapping and viscosity. In contrast, physics maintenance and state transformation is focused more on the content of the model. Other challenges in reasoning might relate to the ability to contextualize the system within a larger environment or the distribution of system-level functions to several distributed elements.

6. BENCHMARK PROBLEM TYPES

In order to explore the different characteristics of representation that enable cognitive criteria to support reasoning activities, a set of benchmark problems are needed. Expanding upon the problem type classifications in the literature, we propose four types for study, an example for each type found in the literature, and a list of alternative examples for each. We do not include large-scale, complex systems such as submarines, aircraft, or space systems as these are not readily available to all researchers for benchmarking activities. A limitation of many benchmarking efforts is that the problems selected should be relatively simple so that researchers are not dissuaded from applying their approaches against similar problems. Complex systems might be of interest in benchmarking, but the effort involved in constructing these models might be too great when compared to the value of understanding the differences. This challenge is noted here but is not resolved.

6.1. Reverse engineered products

Many function modeling approaches have been demonstrated on existing products after dissection and reverse engineering. A repository of commercial products that have been reversed engineered to understand them has been developed with the function representation serving as the foundation for the information model (Bohm et al., 2005). An advantage of including this type of problem in the benchmarking formalism is that the products exist, and their performance can be measured and evaluated. A reverse engineered product provides a common platform for comparison. An example product that has been used extensively (Maier, Ezhilan, et al., 2007; Huang & Jin, 2009; Hamraz et al., 2012) to explore function modeling is the hairdryer (Figs. 2 and 3). Other possible products that can be considered are pneumatic impact drivers (a greater number of mechanical components), battery power tools (readily available in multiple variations), vacuum cleaners (for comparison across multiple customer cultural differences), bike lights (simple and inexpensive systems), or the glue gun (simple product with material flow) proposed for this Special Issue (Mocko et al., 2007; Summers et al., 2017). The selection of a common product is most important so that the community can standardize their demonstration cases.

6.2. Familiar product

Engineers use reverse engineering and the dissection of products to map existing systems and components to specific functionality, the first step of which involves hypothesizing the internal functioning of a product (Otto & Wood, 1998). Further, while reverse engineering can test the ability of a representation to model the detailed functionality of an existing system, modeling a familiar without the product in hand can expose the ability to be fluidly and flexibly model the system, as significant backtracking and hierarchical jumping is likely. An example of the results from an experimental exercise to explore how engineers model known products is found in Figure 4 (Eckert et al., 2011). Different engineers are likely to model the system in different ways even given the same underlying representation, so the expressive power of modeling approaches can be assessed. Thus, this benchmark product can be used to explore defining characteristics of the representation, such as consistency

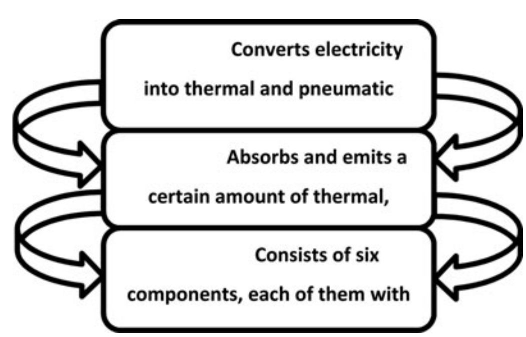

Fig. 2. Example knowledge types in the hairdryer (Hamraz et al., 2012).

https://doi.org/10.1017/S0890060417000476 Published online by Cambridge University Press

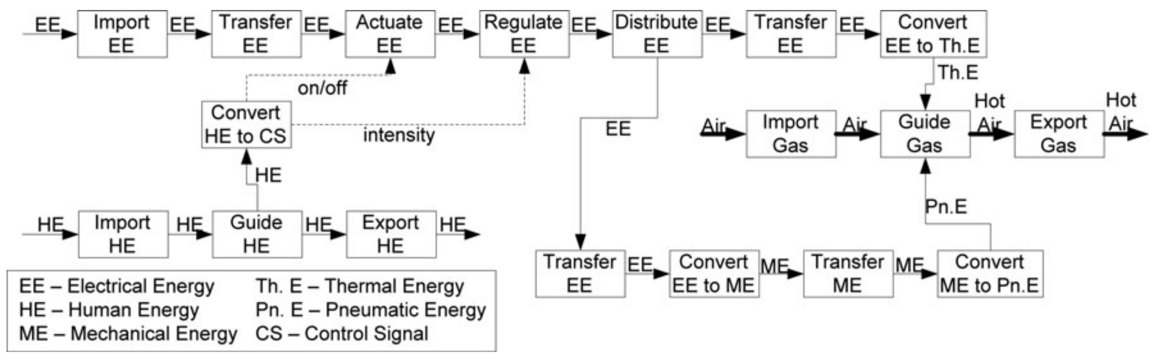

Fig. 3. Function structure of a hairdryer product stored in the Design Repository (http://repository.designengineeringlab.org/).

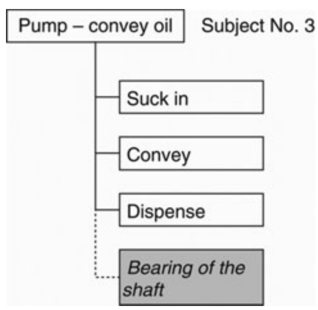

Fig. 4. Example function model for a hydraulic pump (Eckert et al., 2011).

and repeatability, without the cost of buying products to reverse engineer. Other products considered for reverse engineering are bicycles, gear boxes, or printing machines.

6.3. Novel products

Generative forward system design involves the development of new multicomponent systems for characteristics not previously addressed: novelty (not yet attempted), system (multiple components), and intentional (design with a purpose). Examples of "problems" that used to benchmark and compare different function modeling approaches include automated omelet makers, hand-cranked pretzel makers, shoe-string tying mechanisms, clothes folding machines in hotel laundries, and a hand-cranked automated burrito maker. Benchmark examples can be drawn from literature to support the objectivity of the benchmark. For example, the burrito folding system problem has been used previously in the comparing of function lists and function structures in morphological charts (Richardson et al., 2011). Figure 5 illustrates the function lists and structures for the burrito folder that were used in these ideation experiments. Such a benchmark is useful in exploring the degree to which a function representation can be used in understanding novel problems and generating new solutions.

6.4. Single-component products

Single-component multifunctional products, such as passive morphing airfoils (Schultz et al., 2010) and speed screws are also useful as benchmarks (Albers et al., 2008). For example, the design and analysis of a speed screw demonstrated the effective use of the contact and channel model in the design of a single component. Given that function models cannot capture the functionality and behaviors associated with single components, however, this scaling ability in both small and large systems must be explored. While the speed screw benchmark example (Fig. 6) shows the downward scalability of reverse engineering, the passive, morphing airfoil design (Fig. 7) illustrates the downward scalability of forward engineered products. Larger scale systems, such as aircraft, are not considered within the benchmarking protocol due to the challenge of general access for the researchers.

6.5. Problem characteristics

While these problem types are focused on the "thing" to be modeled within the function representations, other characteristics may also be considered when comparing the problems. This list is not exhaustive and is intended to recommend to the community other criteria to be considered when developing the benchmark problems for experimental studies. Defining

(store, {filling}, {filling})
(position, {tortilla}, {tortilla})
(fill, {filling, tortilla}, {unfolded})
(fold, {unfolded}, {burrito})
(dispense, {burrito}, {burrito})

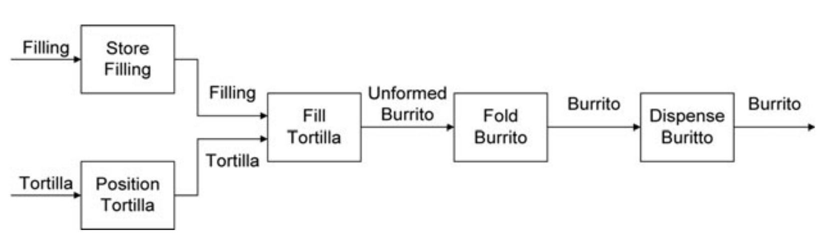

Fig. 5. Function list and function structure for burrito folder (Richardson et al., 2011).

https://doi.org/10.1017/S0890060417000476 Published online by Cambridge University Press

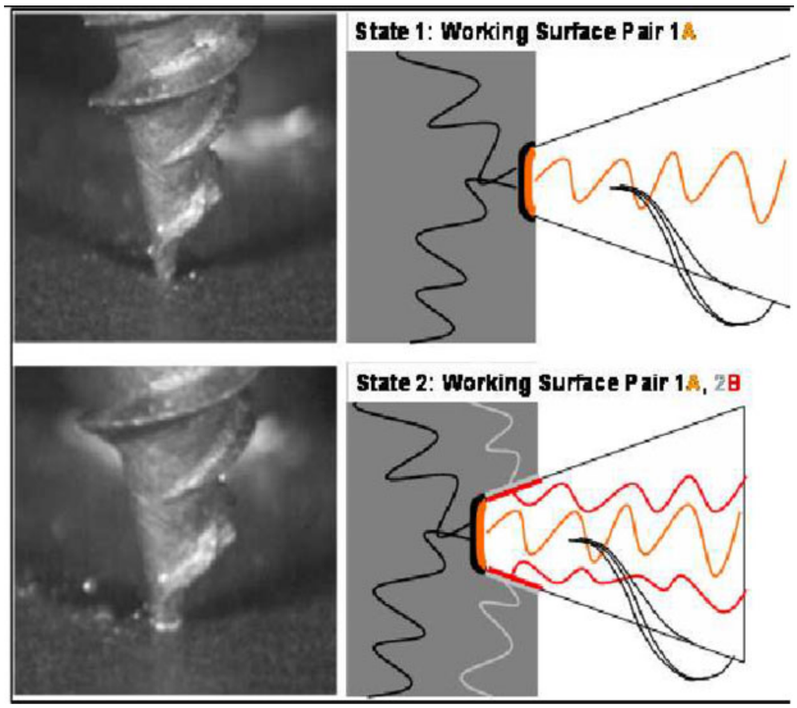

Fig. 6. States of working surface pairs for the contact and channel model for a speed screw (Albers et al., 2008).

these characteristics will enable the development of similar problems to capture and address comparable criteria.

For instance, measures of the problem size, problem connectivity, and problem difficulty are used to compared problem complexity (Summers & Shah, 2010). Similarly, an information metric was used to compare size complexity (i.e., different problems and prototypes) to explore the impact of representation and level of fidelity in assessing requirement satisfaction (Hannah et al., 2011). Finally, problem size as measured with a requirement count and number of words was used to compare design problems in a function modeling experiment (Worinkeng et al., 2015; Patel, Kramer, et al., 2016).

Other considerations used in defining a design problem included the following:

- the domain appropriateness for the participants (Ostergaard et al., 2005; Wetmore et al., 2010; Sen & Summers, 2014);
- realistic and not contrived concepts (Linsey et al., 2010; Thiagarajan et al., 2017);
- constraints and conditions of the problem (Stacey & Eckert, 2010; Eckert & Stacey, 2014);
- a compelling and intrinsic motivation for the participants (Linsey et al., 2010; Joshi & Summers, 2014); and

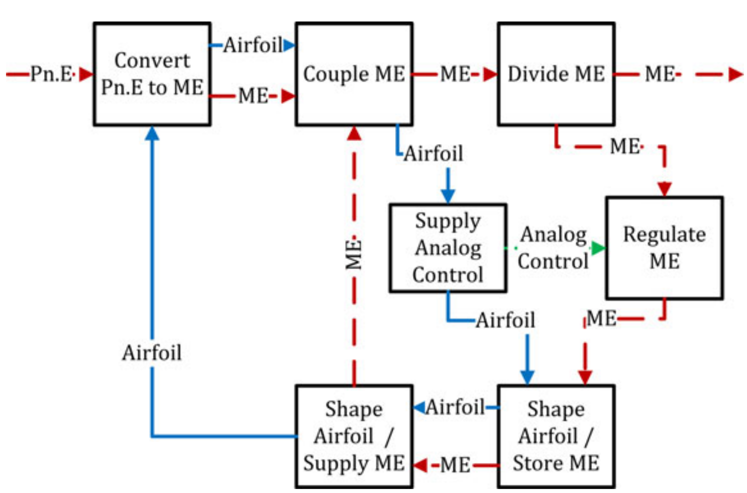

Fig. 7. Function structure for a morphing airfoil (Schultz et al., 2010).

https://doi.org/10.1017/S0890060417000476 Published online by Cambridge University Press

- representation characteristics of both expression and abstraction (McKoy et al., 2001; Summers & Shah, 2004; Hannah et al., 2011).

The authors wish to emphasize that the problems developed should be common across multiple criteria for the cross-comparison between benchmarking results, particularly when using sampling logic for the experimental analysis.

7. RESEARCH RECOMMENDATIONS

While this paper is written as a call to action for the establishment of standards for function modeling benchmarking, such standards should be developed and evolved with input from both users and developers of the various modeling approaches. Such collaborations should also occur in an academically open environment. Researchers will find clear value in this benchmarking process as it forces the disparate communities to begin to communicate with each other, distributes tutorials on the creation and execution of a variety of models and methods to enhance the education of future engineers, and can be paired with a reasoning/representation selection database to systematically develop informed tools and methods. The benchmarking exercises can also help researchers justify a systematic evolution of their approaches.

We do not see benchmarking as an alternative to validation and verification of functional modeling approaches, but as an addition, which can help in the process of evaluating the approach. While validation and verification can attest to the completeness and correctness of a functional modeling approach, benchmarking helps us compare multiple approaches in terms of information captured, reasoning supported, computational efficiency, and cognitive plausibility.

An internationally diverse benchmarking development group representing the various approaches is a recommended strategy for creating this protocol. In addition to the intellectual motivation for benchmarking, this group can pressure both public and private sector agencies to allocate research funds to support the development of benchmarking schemes.

These benchmarking criteria described above are not intended to serve as the final set, but rather to serve as the point of departure. Additional study is needed to refine these characteristics and to strategically select case examples for comparison. This proposed development group can support these endeavors directly.

REFERENCES

Agyemang, M., Turner, C.J., & Linsey, J. (2017). Transforming functional models to critical chain models via expert knowledge and automatic parsing rules for design analogy identification. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 31(4)*, 501–511.

Albers, A., Thau, S., & Alink, T. (2008). Support of design engineering activity through C & CM—temporal decomposition of design problems. *Proc. Tools and Methods for Competitive Engineering Conf.* (Horvath, I., Ed.), pp. 295–306, Izmir, Turkey.

Andreasen, M.M., & Hein, L. (1987). *Integrated Product Development*. London: IFS Publications.

Arlitt, R.M., Stone, R.B., & Tumer, I.Y. (2016). Impacts of function-related research on education and industry. In *Impact of Design Research on Industrial Practice* (Chakrabarti, A., & Lindemann, U., Eds.), pp. 77–99. New York: Springer.

Bhatta, S.R., & Goel, A.K. (1997). A functional theory of design patterns. *Proc. 15th Int. Joint Conf. Artificial Intelligence*, Vol. 1, pp. 294–300, Nagoya, Japan.

Blackwell, A., Britton, C., Cox, A., Green, T., Gurr, C., Kadoda, G., Kutar, M., Loomes, M., Nehaniv, C., Petre, M. (2001). Cognitive dimensions of notations: design tools for cognitive technology. In *Cognitive Technology: Instruments of Mind* (Beynon, M., Nehaniv, C., & Dautenhahn, K., Eds.), LNCS, Vol. 2117, pp. 325–341. Berlin: Springer.

Bohm, M., Stone, R., & Szykman, S. (2005). Enhancing virtual product representations for advanced design repository systems. *ASME Transactions Journal of Computing and Information Science in Engineering 5(4)*, 360–372.

Bracewell, R.H., & Sharpe, J.E.E. (1996). Functional descriptions used in computer support for qualitative scheme generation—"Schemebuilder." *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 10(4)*, 333–345.

Braha, D., & Maimon, O. (1998). The measurement of a design structural and functional complexity. *IEEE Transactions on Systems, Man, and Cybernetics: Part A—Systems and Humans 28(4)*, 527–535.

Brest, J., Greiner, S., Boskovic, B., Mernik, M., & Zumer, V. (2006). Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. *IEEE Transactions on Evolutionary Computation 10(6)*, 646–657. doi:10.1109/TEVC.2006.872133

Bryant, C.R., McAdams, D.A., Stone, R.B., Kurtoglu, T., & Campbell, M.I. (2006). A validation study of an automated concept generator design tool. *Proc. ASME Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, Vol. DTM, pp. 283–294. Philadelphia, PA: ASME.

Buede, D.M., & Miller, W.D. (2016). *The Engineering Design of Systems: Models and Methods*. Hoboken, NJ: Wiley.

Caldwell, B.W., Ramachandran, R., & Mocko, G.M. (2012). Assessing the use of function models and interaction models through concept sketching. *Proc. ASME Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, pp. 633–646, Chicago.

Caldwell, B.W., Thomas, J., Sen, C., Mocko, G.M., & Summers, J.D. (2012). The effects of language and pruning on function structure interpretability. *Journal of Mechanical Design 134(6)*, 61001. doi:10.1115/1.4006442

Campean, I.F., Henshall, E., & Rutter, B. (2013). Systems engineering excellence through design: an integrated approach based on failure mode avoidance. *SAE International Journal of Materials and Manufacturing 6(2013-01–0595)*, 389–401.

Cebrian-Tarrason, D., Lopez-Montero, J.A., & Vidal, R. (2008). OntoFaBeS: ontology design based in FBS framework. *Proc. CIRP Design Conference: Design Synthesis*. Enschede, The Netherlands: CIRP.

Chandrasekaran, B. (2005). Representing function: relating functional representation and functional modeling research streams. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 19(2)*, 65–74.

Chiang, W.-C., Pennathur, A., & Mital, A. (2001). Designing and manufacturing consumer products for functionality: a literature review of current function definitions and design support tools. *Integrated Manufacturing Systems 12(6)*, 430–448. doi:10.1108/EUM0000000006108

Collins, J.A., Hagan, B.T., & Bratt, H.M. (1976). The failure-experience matrix—a useful design tool. *Journal of Engineering for Industry 98(3)*, 1074. doi:10.1115/1.3439009

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

Deng, Y.M. (2002). Function and behavior representation in conceptual mechanical design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 16(5)*, 343–362. Retrieved from doi:10.1017/S0890060402165024

Dolan, E.D., & Moré, J.J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming 91(2)*, 201–213.

Eastman, C.M. (1969). Cognitive processes and ill-defined problems: a case study from design. *Proc. Joint Int. Conf. Artificial Intelligence*, Vol. 69, pp. 669–690, Washington, DC.

Eckert, C. (2013). That which is not form: the practical challenges in using functional concepts in design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 27(3)*, 217–231.

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. doi: 10.1080/09544828.2011.603297

Eckert, C., Ruckpaul, A., Alink, T., & Albers, A. (2012). Variations in functional decomposition for an existing product: experimental results. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 26(2)*, 107–128.

Eckert, C.M., & Stacey, M.K. (2014). Constraints and conditions: drivers for design processes. In *An Anthology of Theories and Models of Design* (Chakrabarti, A., & Blessing, L.T.M., Eds.), pp. 395–415. London: Springer.

Erden, M.S., Komoto, H., van Beek, T.J., D'Amelio, V., Echavarria, E., & Tomiyama, T. (2008). A review of function modeling: approaches and applications. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 22(2)*, 147–169. doi:10.1017/S0890060408000103

Fenves, S.J., Foufou, S., Bock, C., & Sriram, R.D. (2008). CPM2: a core model for product data. *Journal of Computing and Information Science in Engineering 8(1)*, 14501.

Freeman, P., & Newell, A. (1971). A model for functional reasoning in design. *Proc. 2nd Int. Joint Conf. Artificial Intelligence*, pp. 1–2, London, September 1–3.

Gaffney, E.S., Maier, J.R.A., & Fadel, G.M. (2007). Roles of function and affordance in the evolution of artifacts. *Proc. Int. Conf. Engineering Design, ICED*, pp. 1–12. Paris: Design Society.

Gericke, K., & Eisenbart, B. (2017). The integrated function modelling framework and its relation to function structures. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 31(4)*, 436–457.

Gero, J. (1996). Creativity, emergence, and evolution in design. *Knowledge-Based Systems 9(7)*, 435–448.

Gero, J., & Kannengiesser, U. (2002). The situated-function-behaviour-structure framework. In *Artificial Intelligence in Design* (Gero, J., Ed.), pp. 89–104. Norwell, MA: Kluwer Academic.

Gero, J.S., & Kannengiesser, U. (2004). The situated function-behaviour-structure framework. *Design Studies 25(4)*, 373–391. doi:10.1016/j.destud.2003.10.010

Gill, A., & Summers, J.D. (2016). Impact of level of detail and information content on accuracy of function structure-based market price prediction models. *Proc. ASME Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, Paper No. DETC2016-59662. Charlotte, NC: ASME.

Goel, A.K. (1997). Design, analogy, and creativity. *IEEE Intelligent Systems 12(3)*, 62–70.

Goel, A.K. (2013). One thirty year long case study: fifteen principles: implications of the AI methodology for functional modeling. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 27(3)*, 203–215.

Goel, A.K., & Bhatta, S.R. (2004). Use of design patterns in analogy-based design. *Advanced Engineering Informatics 18(2)*, 85–94.

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.

Green, T. (1991). Describing information artifacts with cognitive dimensions and structure maps. In *People and Computers VI, Proc. HCI'91* (Diaper, D., & Hammond, N., Eds.), pp. 297–315. New York: Cambridge University Press.

Green, T., & Petre, M. (1996). Usability analysis of visual programming environments: a "cognitive dimensions" framework. *Journal of Visual Languages and Computing 7(2)*, 131–174.

Hamraz, B., Caldwell, N.H.M., & Clarkson, P. (2012). A multidomain engineering change propagation model to support uncertainty reduction and risk management in design. *Journal of Mechanical Design 134(10)*, 100905.

Hannah, R., Joshi, S., & Summers, J.D. (2011). A user study of interpretability of engineering design representations. *Journal of Engineering Design 23(6)*, 443–468. doi:10.1080/09544828.2011.615302

Hirtz, J., Stone, R.B., McAdams, D.A., Szykman, S., & Wood, K.L. (2002). A functional basis for engineering design: reconciling and evolving previous efforts. *Research in Engineering Design 13(2)*, 65–82.

Huang, Z., & Jin, Y. (2009). Extension of stress and strength interference theory for conceptual design-for-reliability. *Journal of Mechanical Design 131(7)*, 71001.

Hubka, V., & Eder, W.E. (1988). *Theory of Technical Systems: A Total Concept Theory for Engineering Design*, Vol. 1. New York: Springer Verlag.

Joshi, S., & Summers, J.D. (2014). Impact of requirements elicitation activity on idea generation: a designer study. *Proc. ASME* 2014 *Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, pp. V007T07A026-V007T07A026. New York: ASME.

Kirschman, C.F., & Fadel, G.M. (1998). Classifying functions for mechanical design. *Journal of Mechanical Design 120(3)*, 475–482. doi:10.1115/1.2829176

Kurfman, M.A., Stock, M.E., Stone, R.B., Rajan, J., & Wood, K.L. (2003). Experimental studies assessing the repeatability of a functional modeling derivation method. *Journal of Mechanical Design 125(4)*, 682–693.

Kurtoglu, T., & Tumer, I. (2008). A graph-based fault identification and propagation framework for functional design of complex systems. *ASME Transactions Journal of Mechanical Design 130(5)*, 1–8.

Linsey, J.S., Tseng, I., Fu, K., Cagan, J., Wood, K.L., & Schunn, C. (2010). A study of design fixation, its mitigation and perception in engineering design faculty. *Journal of Mechanical Design 132(4)*, 41003.

Linz, P. (2011). *An Introduction to Formal Languages and Automata*, 5th ed. Sudberry, MA: Jones & Bartlett.

Lucero, B., Viswanathan, V.K., Linsey, J.S., & Turner, C.J. (2014). Identifying critical functions for use across engineering design domains. *Journal of Mechanical Design 136(12)*, 121101.

Maier, J.R.A., Ezhilan, T., Fadel, G.M., Summers, J.D., & Mocko, G. (2007). A hierarchical requirements modeling scheme to support engineering innovation. *Proc. Int. Conf. Engineering Design*, pp. 839–840. Paris: Design Society.

Maier, J.R.A., Srinivasan, A., Vikram, B., Summers, J.D., & Bernhard, B. (2007). A computational framework for semantically rich design problems based on the theory of affordances and examplar technology. *Proc. Int. Conf. Engineering Design*, pp. 205–206. Paris: Design Society.

Mathieson, J.L., Arlitt, R., Summers, J.D., Stone, R., Shanthakumar, A., & Sen, C. (2011). Complexity as a surrogate mapping between function models and market value. *Proc. ASME Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, Paper No. DETC2011–47481. Washington, DC: ASME.

McKoy, F.L., Vargas-Hernández, N., Summers, J.D., & Shah, J.J. (2001). Influence of design representation on effectiveness of idea generation. *Proc. ASME Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.* Pittsburgh, PA: ASME.

Mocko, G.M., Summers, J.D., Teegavarapu, S., Ezhilan, T., Maier, J.R.A., & Fadel, G.M. (2007). A modelling scheme for capturing and analyzing conceptual desing information: an application to the hair dryer example and comparison to existing literature. *Proc. Int. Conf. Engineering Design*, Paper No. DS42_P_570. Paris: Design Society.

Mohinder, C.V.S.R., Gill, A., & Summers, J.D. (2017). Using the graph complexity connectivity method to predict information from design representations: a comparative study. *Proc. Design Computing and Cognition'*16, pp. 667–683. Berlin: Springer.

Mokhtarian, H., Coatanea, E., & Henri, P. (2017). Function modeling combined with physics-based reasoning for assessing design options and supporting innovative ideation. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 31(4)*, 476–500.

Nagel, R.L., Bohm, M.R., Linsey, J.S., & Riggs, M.K. (2015). Improving students' functional modeling skills: a modeling approach and a scoring rubric. *Journal of Mechanical Design 137(5)*, 51102.

Nagel, R.L., Perry, K.L., Stone, R.B., & McAdams, D.A. (2009). Function-CAD: a functional modeling application based on the function design framework. *Proc. ASME Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, pp. 591–600. San Diego, CA: ASME.

Nagel, R.L., Stone, R.B., Hutcheson, R.S., McAdams, D.A., & Donndelinger, J.A. (2008). Function design framework (FDF): integrated process and function modeling for complex systems. *Proc. Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, pp. 273–286. Brooklyn, NY: ASME.

Nagel, R.L., Vucovich, J.P., Stone, R.B., & McAdams, D.A. (2008). A signal grammar to guide functional modeling of electromechanical products. *Journal of Mechanical Design 130(5)*, 51101.

Nebel, B. (2000). On the compilability and expressive power of propositional planning formalisms. *Journal of Artificial Intelligence Research 12(1)*, 271–315.

Olewnik, A.T., & Lewis, K. (2005). On calidating engineering design decision support tools. *Concurrent Engineering 13(2)*, 111–122.

Ostergaard, K.J., Wetmore, W. R., III., Divekar, A., Vitali, H., & Summers, J.D. (2005). An experimental methodology for investigating communication in collaborative design review meetings. *Co-Design 1(3)*, 169–185.

Otto, K.N., & Wood, K.L. (1998). Product evolution: a reverse engineering and redesign methodology. *Research in Engineering Design 10(4)*, 226–243.

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., Blessing, L., Feldhusen, J., Grote, K.-H.H., & Wallace, K. (2013). *Engineering Design: A Systematic Approach*, 3rd ed., Vol. 11. London: Springer-Verlag.

Patel, A., Andrews, P., & Summers, J.D. (2016). Evaluating the use of artificial neural networks to predict assembly defects. *Proc. Int. Design Engineering Conf. Computers in Engineering Conf. (ASME IDETC/CIE)*, Paper No. DETC2016-59664. Charlotte, NC: ASME.

Patel, A., Kramer, W., Summers, J.D., & Shuffler-Porter, M. (2016). Function modeling: a study of sequential model completion based on count and chaining of functions. *Proc. ASME Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, Paper No. DETC2016-59860. Charlotte, NC: ASME.

Peterson, C. (1990). Parallel distributed approaches to combinatorial optimization: benchmark studies on traveling salesman problem. *Neural Computation 2(3)*, 261–269.

Qian, L., & Gero, J.S. (1996). Function-behavior-structure paths and their role in analogy-based design. *Artificial Intelligence for Engineering, Design, Analysis and Manufacturing 10(4)*, 289–312.

Rajamani, R. (2012). Roll dynamics and rollover prevention. In *Vehicle Dynamics and Control* (Ling, F.F., Ed.), pp. 427–456. Boston: Springer.

Regli, W.C., Hu, X., Atwood, M., & Sun, W. (2000). A survey of design rationale systems: approaches, representation, capture and retrieval. *Engineering With Computers 16(3)*, 209–235.

Richardson, J.L., III, Summers, J.D., & Mocko, G.M. (2011). Function representations in morphological charts: an experimental study on variety and novelty on means generated. *Interdisciplinary Design: Proc. 21st CIRP Design Conf.*, pp. 76–84, Daejeon, South Korea.

Rodenacker, W. (1971). *Methodisches Konstruieren*. Berlin: Springer-Verlag.

Rosenman, M.A., & Gero, J.S. (1998). Purpose and function in design: from the socio-cultural to the techno-physical. *Design Studies 19(2)*, 161–186.

Sasajima, M., Kitamura, Y., Ikeda, M., & Mizoguchi, R. (1995). FBRL: a function and behavior representation language. *Proc. Int. Joint Conf, Artificial Intelligence*, Montreal.

Schultz, J., Sen, C., Caldwell, B., Mathieson, J., Summers, J.D., & Mocko, G.M. (2010). Limitations to function structures: a case study in morphing airfoil design. *Proc. International Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, Paper No. DETC2010-28559. Montreal: ASME.

Sembugamoorthy, V., & Chandrasekaran, B. (1986). Functional representation of devices and compilation of diagnostic problem-solving systems. In *Experience, Memory, and Reasoning* (Kolodner, J., & Riesbeck, C.K., Eds.), pp. 47–53. Hillsdale, NJ: Erlbaum.

Sen, C., & Summers, J.D.J. (2014). A pilot protocol study on how designers construct function structures in novel design. *Proc. Design Computing and Cognition'12* (Gero, J., Ed.), pp. 247–264. College Station, TX: Springer-Verlag.

Sen, C., Summers, J.D., & Mocko, G.M. (2011*a*). A protocol to formalise function verbs to support conservation-based model checking. *Journal of Engineering Design 22(11–12)*, 765–788.

Sen, C., Summers, J.D., & Mocko, G.M. (2011*b*). Exploring potentials for conservational reasoning using topologic rules of function structure graphs. *Proc. 18th Int. Conf. Engineering Design (ICED11)*, Vol. 9, pp. 377–388, Copenhagen, Denmark.

Sen, C., Summers, J.D., & Mocko, G.M. (2013*a*). A formal representation of function structure graphs for computer-directed modeling and conservation-based reasoning. *ASME Transactions Journal of Computing and Information Science in Engineering 13(2)*, 21001.

Sen, C., Summers, J.D., & Mocko, G.M. (2013*b*). A formal representation of function structure graphs for physics-based reasoning. *Journal of Computing and Information Science in Engineering 13(2)*, 21001.

Sen, C., Summers, J.D., & Mocko, G.M. (2013*c*). Physics-based reasoning in conceptual design using a formal representation of function structure graphs. *Journal of Computing and Information Science in Engineering 13(1)*, 11008.

Shah, J.J., & Mantyla, M. (1995). *Parametric and Feature-Based CAD/CAM: Concepts, Techniques, and Applications*. New York: Wiley.

Smith, G., Richardson, J., Summers, J.D., & Mocko, G.M. (2012). Concept exploration through morphological charts: an experimental study. *Journal of Mechanical Design 134(5)*, 051004.1-10.

Sridhar, S., Fazelpour, M., Gill, A.S., & Summers, J.D. (2016). Accuracy and precision analysis of the graph complexity connectivity method. *Procedia CIRP 44*, 163–168.

Srinivasan, V., Chakrabarti, A., & Lindemann, U. (2012). A framework for describing functions in design. *Proc. Int. Design Conf.*, pp. 1111–1122. Dubrovnik, Croatia.

Stacey, M., & Eckert, C. (2010). Reshaping the box: creative designing as constraint management. *International Journal of Product Development 11(3–4)*, 241–255.

Stone, R.B., Tumer, I.Y., & Van Wie, M. (2005). The function-failure design method. *ASME Transactions Journal of Mechanical Design 127(3)*, 397–407.

Sturges, R.H., O'Shaughnessy, K., & Kilani, M.I. (1996). Computational model for conceptual design based on extended function logic. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 10(4)*, 255–274.

Suh, N. (1999). A theory of complexity, periodicity, and the design axioms. *Research in Engineering Design 11(2)*, 116–131.

Summers, J.D., Eckert, C.M., & Goel, A.K. (2017). Function in engineering: benchmarking representations and models. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 31(4)*, 401–412.

Summers, J.D., & Rosen, D.W. (2013). Mechanical engineering modeling language: foundations. *Proc. Int. Conf. Engineering Design*. Seoul, South Korea: Design Society.

Summers, J.D., & Shah, J.J. (2004). Representation in engineering design: a framework for classification. *Proc. Int. Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, Paper No. DTM-57514. Salt Lake City, UT: ASME.

Summers, J.D., & Shah, J.J. (2010). Mechanical engineering design complexity metrics: size, coupling, and solvability. *Journal of Mechanical Design 132(2)*, 21004.

Telenko, C., & Seepersad, C.C. (2010). A methodology for identifying environmentally conscious guidelines for product design. *Journal of Mechanical Design 132(9)*, 91009.

Thiagarajan, A., Patel, A., O'Shields, S., & Summers, J.D. (2017). Functional thinking: a protocol study to map modeling behavior of designers. *Proc. Design Cognition and Computing (DCC16)* (Gero, J.S., Ed.), pp. 339–357. Evanston, IL: Springer.

Tomiyama, T., Van Beek, T.J., Cabrera, A.A.A., Komoto, H., & D'Amelio, V. (2013). Making function modeling practically usable. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 27(3)*, 301–309.

Ullman, D.G. (2010). *The Mechanical Design Process*, 4th ed. New York: McGraw-Hill.

Ullman, D.G., Dietterich, T.G., & Stauffer, L.A. (1988). A model of the mechanical design process based on empirical data. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 2(1)*, 33–52.

Ulrich, K., & Eppinger, S.D. (2008). *Product Design and Development*, 4th ed. New York: McGraw-Hill.

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.

Vargas-Hernandez, N., & Shah, J.J. (2004). 2 nd-CAD: a tool for conceptual systems design in electromechanical domain. *Journal of Computing and Information Science in Engineering 4(1)*, 28–36.

Vermaas, P.E. (2013). On the co-existence of engineering meanings of function: four responses and their methodological implications. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 27(3)*, 1–19.

Vescovi, M., Iwasaki, Y., Fikes, R., & Chandrasekaran, B. (1993). CFRL: a language for specifying the causal functionality of engineered devices. *Proc. 11th National Conf. Artificial Intelligence*. Washington, DC: American Association for Artificial Intelligence.

Wang, C., Qing-liang, Z., & Albers, A. (2009). A unified product model based on C&CM for mechatronic product development. *Proc. ICMA Int. Conf. Mechatronics and Automation*, pp. 2173–2177. New York: IEEE.

Wetmore, W.R., III, Summers, J.D., & Greenstein, J.S. (2010). Experimental study of influence of group familiarity and information sharing on design review effectiveness. *Journal of Engineering Design 21(1)*, 111–126.

Wiltgen, B., & Goel, A. (2016). Functional model simulation for evaluating design concepts. *Advances in Cognitive Systems*. Advance online publication.

Winston, P.H. (2005). *Artificial Intelligence*, 4th ed. Reading, MA: Addison-Wesley.

Worinkeng, E., Joshi, S., & Summers, J.D. (2015). An experimental study: analyzing requirement type influence on novelty and variety of generated solutions. *International Journal of Design Creativity and Innovation 3(2)*, 61–77.

Yang, S., Patil, L., & Dutta, D. (2010). Function semantic representation (FSR): a rule-based ontology for product functions. *ASME Journal of Computing and Information Science in Engineering 10*, 31001.

Yildirim, U., Campean, F., & Williams, H. (2017). Function modelling using the system state flow diagram. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 31(4)*, 413–435.

Joshua D. Summers is a Professor of mechanical engineering at Clemson University, where he is also the Co-Director of the Clemson Engineering Design Applications and Research Group. He earned his PhD in mechanical engineering from Arizona State University and his MS and BS the University of Missouri. Dr. Summers worked at the Naval Research Laboratory (VR Lab and NCARAI). His research has been funded by government, large industry, and small- to medium-sized enterprises. Joshua's areas of interest include collaborative design, knowledge management, and design enabler development with the overall objective of improving design through collaboration and computation.

Claudia Eckert is a Professor of design at Open University. She has a longstanding interest in studying and supporting industrial practice in different design domains and has published numerous papers on it. In particular, she has been working on process modeling, engineering change, and functional modeling of complex engineering products.

Ashok K. Goel is a Professor of computer science in the School of Interactive Computing at Georgia Institute of Technology. He is the Director of the School's PhD program in Human-Centered Computing and the Design & Intelligence Laboratory. Ashok is also Co-Director of the Institute's Center for Biologically Inspired Design and a Fellow of the Brooke Byer's Institute for Sustainable Systems. He served on the board of directors of the Biomimicry Institute from 2012 to 2017 and was its President from 2015 to 2017. Dr. Goel has been conducting research into artificial intelligence, cognitive science, and human-centered computing for 30 years, with a focus on computational design, modeling, and creativity. His 2012 TEDx@Peachtree talk summarized some of his research on computational design and creativity. An interactive tool for supporting some aspects of biologically inspired design was developed by his laboratory. Ashok is Editor of AAAI's *AI Magazine* and an Associate Editor of the Design Research Society's *Design Science Journal* as well as *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*.