

The integrated function modeling framework and its relation to function structures

KILIAN GERICKE¹ AND BORIS EISENBART²

¹Research Unit in Engineering Sciences, University of Luxembourg, Belval, Luxembourg

²School of Design, Swinburne University of Technology, Melbourne, Australia

(RECEIVED November 16, 2016; ACCEPTED May 30, 2017)

Abstract

Research and industrial practice have produced a host of function models and modeling approaches over the last decades. Each of these is meant to support designers in their design endeavors. Industrial practice is excessively diversified in terms of contextual requirements, aims, and adopted processes; this automatically begs the question which of the existing models should be selected for application in a specific situation. This paper sets out to contribute to this discourse. It strives to benchmark the fairly novel integrated function modeling (IFM) framework against the well-established function structures modeling approach. The paper comparatively investigates the respective capabilities of the approaches, following the benchmarking protocol used earlier in relation to this Special Issue. Function structures are used as reference as they represent one of the most widespread function modeling approaches in research and practice. Both function structures and the IFM framework are exemplarily applied for modeling a glue gun. The gradual generation and refinement of the models is used to showcase their respective benefits and shortcomings. Eventually, the IFM framework is found to excel over function structures in terms of comprehensiveness and support for different types of function analyses. Finally, future research directions are proposed.

Keywords: Benchmarking; Conceptual Design; Flexibility; Function Modeling; Functional Perspectives

1. INTRODUCTION

Design research proposes function modeling as a means to support the development of early design concepts (see, e.g., Chakrabarti & Bligh, 2001; Crilly, 2010; Eisenbart et al., 2016a, for an overview). Function modeling is commonly used to describe a system at an abstract level in terms of its distinct effectual causes and relationships, in order to achieve a desired overall behavior, purpose, or task (cf. Warell, 1999; Chiang et al., 2001; Maier & Fadel, 2001; Chandrasekaran, 2005; Ericson & Larsson, 2005; Crilly, 2010; Aurisicchio et al., 2011). Scholars highlight its potential to facilitate concept development through providing means for the representation of the numerous required functions and their dependencies. Furthermore, function modeling helps to ensure all these functions are addressed in the design process and provides a more systematic approach for gradual solution space exploration

and attendant solution synthesis (Blessing & Upton, 1997; US Department of Defense, 2001; Gausemeier et al., 2009; Caldwell et al., 2011; Sen et al., 2013; Tomiyama et al., 2013). Later on in the process, it can support the analysis of function and requirement fulfillment by the design concept. This latter benefit is valuable not only for emergent design solutions but also even more so when making sense of and trying to adapt an existent design, such as in redesign or variant design projects (Maher & Tang, 2003; Far & Elamy, 2005; Andreasen et al., 2015; Eisenbart, Gericke, et al., 2015). More recently, researchers have started discussing an additional benefit of function modeling, namely, that it assists in establishing a shared understanding between collaborating designers from different disciplines, as it is intrinsically abstract and thus discipline unspecific. Erden et al., in particular, suggest that “the barriers between . . . disciplines can be overcome by using [a] common language of functionality” (2008, p. 147, see also Tukker et al., 2006; Müller et al., 2007; van Beek & Tomiyama, 2009). There is initial empirical evidence for the suggested benefits of function modeling to engineering design practice (see Kurfman et al., 2001, 2003; Caldwell et al., 2011; Srinivasan et al., 2012; Sen et al.,

Reprint requests to: Kilian Gericke, Engineering Design and Methodology Group, Research Unit in Engineering Sciences, Faculty of Science, Technology and Communication, University of Luxembourg, Campus Belval, 6, avenue de l'Université, L-4362 Belval, Luxembourg. E-mail: Kilian.Gericke@uni.lu

2013). Yet scholars have noted that function modeling has not been widely taken up in industry (see, e.g., Aurisicchio et al., 2012; Tomiyama et al., 2013). The very characteristic that makes it applicable across disciplinary barriers, that is, its abstract nature, seems to also raise doubts about its practical applicability, deterring companies or individual designers from adopting it in their daily practices (as suggested by Tomiyama et al., 2009; Booth et al., 2015; Eisenbart et al., 2016b). Moreover, existing function modeling approaches incorporate different levels of abstraction and viewpoints in their descriptions of functionality (see Erden et al., 2008; Crilly, 2010; Eisenbart et al., 2013a, for an overview). The differences provide evidence of how versatile the (intended) utilization of function modeling can be. We agree with various scholars contributing to this Special Issue that each function model has its particular advantages and shortcomings. Industrial practice is excessively diversified in terms of contextual requirements, aims, and adopted processes; this automatically begs the question which of the extant models should be selected for application in a specific situation. Here is where benchmarking between function modeling approaches can provide a vital contribution to designers in making a well-founded choice on which model(s) to select (Summers et al., 2013).

This article contributes to this discussion in two ways. First, we explain a new function modeling approach, the integrated function modeling (IFM) framework, which specifically aims to integrate and interlink a large part of the diverse contents and concepts of function modeling from different fields of study (see Section 2). Second, in Section 4, we comparatively investigate the IFM framework's respective capabilities in relation to function structures (as proposed by Pahl et al., 2007). These are used as reference for this analysis as they represent one of the most widespread function modeling approaches in research and practice. Both function structures and the IFM framework are exemplarily applied for function modeling of a glue gun. The specific application of the function models in the given example is compared with respect to the benchmarking protocol by Summers et al. (2013), which has been used earlier in relation to this Special Issue, in order to enhance comparability. Section 5 discusses the models, their strengths and weaknesses, and implications for their use in practice. The identified characteristics of the IFM framework will be used to guide future work that aims to improve the framework and its uptake by industry. Section 7 concludes the paper and suggests implications for further advancement of this research.

2. FUNCTION MODELING

Research has produced a plethora of function models over the years (see Erden et al., 2008; Crilly, 2010; Eisenbart et al., 2013a; Crilly, 2015, for an overview). However, one of the most common representations is the use of verb/noun combinations in relation to flows of operands (i.e., the nouns) and their transformation (the verbs) between the input and the

output of a system. This is incorporated, for instance, in the widespread function structure after Pahl et al. (2007, see Fig. 1).¹

The underlying paradigm has been widely adopted in mechanical engineering literature (see, e.g., Roozenburg & Eekels, 1995; Stone & Wood, 2000; Ulrich & Eppinger, 2008; Ullman, 2010) and in certain interdisciplinary design approaches (see, e.g., VDI, 2004; Cross, 2008; Gausemeier et al., 2009). Function modeling in disciplines like software development, service development, and product-service system (PSS) design predominantly builds on the flow in time (instead of operands) in order to depict procedural relations between individual functions and/or service activities (see, e.g., Shostack, 1982; US Department of Defense, 2001; Kruchten, 2004; Weilkiens, 2008; Watanabe et al., 2011). Electrical engineering, as a final example, tends to focus on distinct states and their transitions to deduce their logical de-velopment and implementation in the final design (see, e.g., Dewey, 2000; Scheffer et al., 2006). What this shows is that function modeling is fairly diverse and can vary quite noticeably between disciplines, especially when it comes to the content of each approach (discussed in detail by Eisenbart et al., 2013a).

Scholars such as Erden et al. (2008), Vermaas (2011, 2013), Carrara et al. (2011), and Garbacz et al. (2011) argue that the multifariousness of function models in itself indicates the vast interest of the research community in the topic. Due to this diversity, however, finding a common theorization of what function and function modeling really entails in design is no easy task. Such a variety of definitions of function can be found in the literature that we will focus here on drawing upon prior work comparing them (see, e.g., Warell, 1999; Chandrasekaran & Josephson, 2000; Chiang et al., 2001; Maier & Fadel, 2001; Deng, 2002; Chandrasekaran, 2005; Ericson & Larsson, 2005; Vermaas, 2009; Crilly, 2010; Aurisicchio et al., 2011; Carrara et al., 2011; Goel, 2013; Eisenbart et al., 2016a, for comprehensive discussions of the different meanings of function conveyed in engineering design). Many of these scholars sought to find a common denominator in extant definitions of function in an attempt to unify or relate between them, respectively. More recently, however, scholars have started to accept or even embrace that function “is a term that has a number of co-existing meanings, which are used side-by-side in engineering” (Vermaas, 2011, p. 98, see also Carrara et al., 2011). Each definition has its specific place in design insofar as it relates to its originators' *interpretation of and viewpoint on* function as a concept (cf. Chittaro & Kumar, 1998; Chandrasekaran & Josephson, 2000; Deng, 2002; Chandrasekaran, 2005). Vermaas (2009, 2013) goes on to derive a set of three notions of function that he considers

¹ Other prominent approaches include the function-behavior-structure framework (Gero, 1990), the structure-behavior-function model (Iwasaki et al., 1993), the function-behavior-state model (Umeda & Tomiyama, 1997), and the conglomerate approaches by Tjalve (1978) or Hubka and Eder (1988; Eder & Hosnedl, 2008).

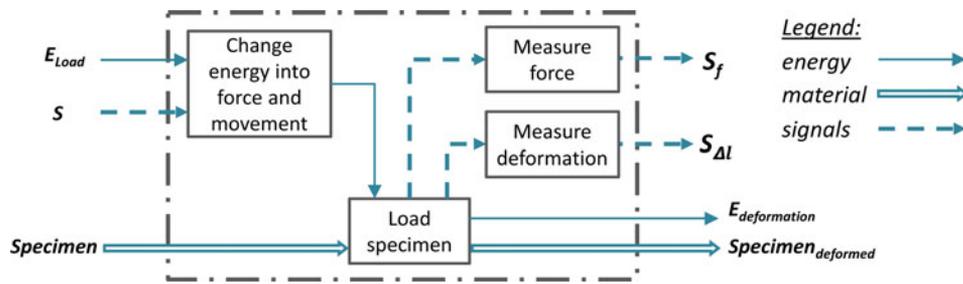


Fig. 1. Schema of function structures after Pahl et al. (2007).

to be archetypical and present in some way in any (formal) function modeling or reasoning approach:

1. *behavior-related notion*: function as the intended behavior of an entity
2. *outcome-related notion*: functions as the desired effects of the behavior of an entity
3. *task- or goal-related notion*: function as the purpose for which an entity is designed

An entity’s behavior may serve a specific, envisaged purpose. Yet, in many cases, it may contribute to entirely different, unforeseen use plans (cf. Houkes & Vermaas, 2010), which are related to so-called affordances (cf. Maier & Fadel, 2001). Affordances cover the entirety of uses that a system can be put to due to its specific characteristics (after Weber, 2007), though these uses may not have been originally intended as such by the designers (Brown & Blessing, 2005).

We follow the idea of Vermaas and others that different views or notions of what function is exist side by side. We already discussed how existing function models vary in terms of content and morphology and thereby reflect coexisting notions and viewpoints. Although this variety is well known, practitioners cannot take advantage of this, as the respective advantages of these modeling approaches have not been explicated. This paper contributes to this gap, by comparing function structures after Pahl et al. (2007) with the IFM framework. Function structures are considered to be well known and will therefore not be described in detail here. The IFM framework is still relatively new and is thus briefly described in the following.

3. THE IFM FRAMEWORK

The IFM framework draws from comprehensive reviews of existing function modeling approaches found in textbooks and industrial practice (see Eisenbart et al., 2016a). It sets out to contribute to design practice by linking diverse contents addressed in function modeling approaches across disciplines, while allowing for flexible adaptation (augmentation or tailoring). This means that designers can focus on the contents required for a specific design task, while disregarding other contents. Hence, modeling efforts and complexity can be adapted to the requirements of a specific project. Specific

adaptations to match different needs in function modeling are discussed by Eisenbart et al. (2013b) from an ontological viewpoint.

The IFM framework is a representational approach that is set up as a combination of modular dependency structure matrices (see Kreimeyer & Lindemann, 2011) and flow modeling. An overview of the framework is provided in Figure 2. In its kernel, the framework consists of adjacently placed and inherently linked views. A specific instance of the IFM framework is called an integrated function (IF) model. An IF model can consist of either all views described by the IFM framework or a subset thereof, depending on which specific entities comprising the framework are relevant in a concrete modeling endeavor. The different views are briefly described in Table 1.

The framework is centered around the *process flow view* (see Fig. 2), which presents the qualitative flow of different types of processes. This was found to be a central representational approach cutting across many disciplinary boundaries in function modeling; it thus serves as a vantage point for an integrative modeling approach (see Eisenbart et al., 2013a). Such processes can be transformational (which is, for instance, intrinsic to function structures after Pahl et al., 2007, and other modeling approaches), but also address interaction processes, human activities, and other major aspects of software, services, and related disciplines, as discussed before. The remaining views complement a function model generated with the IFM framework and eventually arrive at a compre-

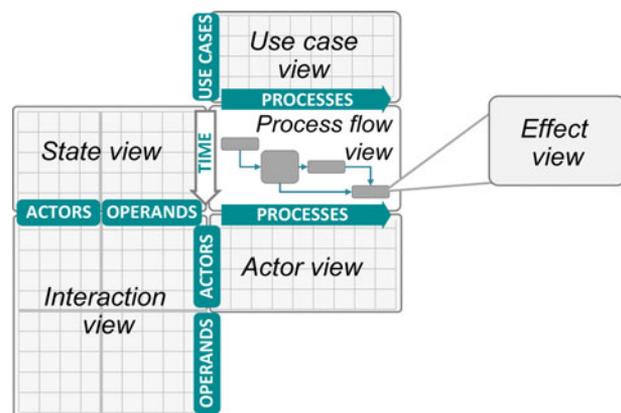


Fig. 2. Schema of the integrated function modeling framework.

Table 1. Associated views in the IFM framework

Process flow view	Qualitatively visualizes the flow of sequential or parallel (interaction and/or transformation) processes related to a specific use case. For each use case an associated set of views is created. In the vertical direction, the process flow is visualized related to time. This matches the flow of states in the associated state view. In the horizontal direction, the process blocks are spread from left to right to enable a direct link to the actor view. Process blocks can indicate parallel and/or partially overlapping execution in time, while arrows between them indicate logical sequences and/or optional paths in the time flow.
State view	Represents the states from initial to final of operands and actors as well as their changes associated with the flow of processes. It can also be indicated if an operand or actor merely supports a process without changing its own state.
Actor view	Indicates the involvement of one or more actors in the realization of individual processes related to a use case. Involvement may be active or passive. Actors can be differentiated based on whether they, from the designers' point of view, are part of the system, which further separates transformation from interaction processes (realized "cross-boundary").
Use case view	Indicates the involvement of individual processes in different use cases. It is intended to support the analysis of dependencies between processes, which may hinder their parallel or sequential execution and thus the operability of use cases in which they are involved.
Effect view	Represents the effects, which enable individual processes and are provided by actors. For each process block in the process flow view, a separate effect view may be created. This allows for detailed analysis of the basic physiochemical effects affecting or contributing to the individual processes.
Interaction view	Uses a combination of matrices that map the specific bilateral impacts between actors and operands as well as their contributions (or any other kind of dependency between them) in the realization of use cases, associated processes, etc. In addition, information about the embodiment of specific bilateral impacts may be included. Hence, this view essentially results in an initial system structure or interface matrix of the system.

Note: Adapted from Eisenbart, Gericke, et al. (2015).

hensive description of the functionality of a modeled entity. Detailed descriptions of the framework and its practical application are provided in Eisenbart et al. (2013a, 2016a).

The entities encompassed in the IFM framework comprise *use cases, transformation and interaction processes, effects, states, operands, and actors*. Use cases represent different scenarios of applying the technical system for a specific purpose (e.g., fulfilling a goal, changing the state of the system or user). For every relevant use case, a new set of views is created. Transformation processes specify *technical* and/or *human* processes that (may) cause a change of state of operands or actors. Therein, processes rely on their underlying physiochemical effects. Operands (as with the Pahl et al., 2007, approach) are specifications of energy, material, and signals. Actors comprise stakeholders (referring to any human or other animate being), technical (sub)systems (which may be hardware and/or software), and parts of the environment that actively or passively contribute to function fulfillment. Finally, interaction processes describe "cross-boundary" interactions between different actors jointly contributing to function fulfillment (see Eder & Hosnedl, 2008).

Through its setup, the framework is intended to accommodate the perceiving function as related to the three archetypical notions of function derived by Vermaas (see above). At its core, the framework centers around the representation of processes and their flows (see Fig. 2), incorporating the notion of function as the (intended) behavior of a system. Explicit visualization of the states of operands and actors directly relates to the notion of function as the effects of an exhibited behavior by virtue of linking these to the associated transformation and interaction processes and the specific changes resulting from them. Finally, use cases intrinsically derive from the distinct applications that a system is put to in relation to users and their respec-

tive use plan or scenario, which corresponds to the notion of function as the purpose of a system or artifact (cf. Eder & Hosnedl, 2008; Houkes & Vermaas, 2010; Vermaas, 2013).

4. BENCHMARKING STUDY DESIGN

Most function modeling approaches are meant to support different types of design tasks, for example, original design, evolutionary design, or reverse engineering (Summers et al., 2013). The information captured in the models can facilitate different activities along the product life cycle (Gericke, Qureshi, et al., 2013). Therefore, benchmarking of function modeling approaches is dependent on the design task and the purpose of its creation and utilization. Summers et al. (2013) provide a protocol for benchmarking of function modeling approaches and propose a categorization of related classification problems by precedence. The proposed protocol aims to support benchmarking of modeling approaches rather than function models (i.e., specific instances of the modeling approaches), and thus supports systematic identification of conceptual improvement potentials in relation to function modeling approaches, and meta-analysis of multiple benchmarks.

The benchmarking study presented in the following aims at comparing two function modeling approaches: function structures as described in Pahl et al. (2007) and the IFM framework introduced earlier.

4.1. Benchmarking problem

Design problems can differ in many dimensions such as complexity, novelty, interdependence, and contradictions of goals and requirements (see Pahl et al., 2007; Ulrich & Eppinger, 2008; Gericke, Meissner, et al., 2013). In order to allow a

systematic comparison of benchmarking studies, a limited set of criteria is required covering a relevant and sufficiently diverse area of real design problems.

The function modeling challenge that is used for benchmarking is a reverse engineering problem of an existing product of low complexity, which has been studied by other scholars in the past, thus facilitating comparability of the obtained insights. The selected case example is a Pattex Hotmelt Glue Gun (see Fig. 3) similar to the one used in the workshop held at the International Conference on Engineering Design in Milan 2015, which seeded this Special Issue.

The glue gun uses electrical energy to melt a hot-melt glue stick and a mechanical feed mechanism to control the extrusion rate of the melted glue. The temperature of the thermoplastic adhesive (i.e., the glue stick) affects the achieved bonding strength. Once the melted glue solidifies, it reaches its final bonding strength within a few seconds. The primary application of such glue guns is for crafts, arts, or related activities.

4.2. Benchmarking protocol

As a precondition for later meta-studies, the individual studies need to use the same or similar categories for comparison. Therefore, this study uses the benchmarking protocol proposed by Summers et al. (2013), which is explained in Section 4.2.1. This set of criteria is complemented by a comparison of the modeled content (see Section 4.2.2) that is based on the classification proposed by Eisenbart et al. (2016a).

4.2.1. Representation, modeling, cognitive dimension, and reasoning characteristics

Summers et al. (2013) propose a set of criteria covering the following categories: representation characteristics, modeling characteristics, cognitive dimension characteristics, and reasoning characteristics. Representation characteristics cover the used vocabulary and grammar of the modeling approach. Modeling characteristics describe the way a modeler creates a model, for example, considering modeling strategies and tool support. Cognitive dimension characteristics allow the investigation of usability issues of function modeling issues. Cog-

nitive dimension characteristics, used in human–computer interaction research, were adapted to function modelling. Reasoning characteristics expand from cognitive dimension characteristics and model content relating to the ability of function modeling approaches to support contextualization of the modeled system and modeling distribution of system-level functions to multiple elements. Table 2 consolidates the explanations of the characteristics.

4.2.2. Perspectives in function modeling approaches

The reasoning characteristics as defined in Table 2 do not cover all entities and modeling characteristics that have been identified to be represented in function modeling approaches. We will use the original set of reasoning characteristics as proposed by Summers et al. (2013) but expand the analysis with a comparison of types of content covered by the modeling approaches. We use a set of modeling perspectives in this analysis (see Table 3), which are derived from a comprehensive analysis of function modeling approaches across multiple disciplines by Eisenbart et al. (2012, 2013a) building on extant research by Vermaas (2013), Eckert (2013; see also Eckert et al., 2010), Erden et al. (2008), Chandrasekaran (2005), and others. Function modeling perspectives encompass the essence of different (types of) content addressed in function models and modeling approaches in bridging a large variety of disciplines. The IFM framework was developed with the goal to incorporate all modeling perspectives; thus, it will perform well with respect to coverage of them. However, we include the extended list of modeling perspectives, first, to point out more clearly the differences in coverage of both approaches, and second, to showcase that this list is a useful amendment to the protocol of Summers et al. (2013). The perspectives are therefore considered appropriate for the purpose of a comparative analysis as intended here.

4.3. Benchmarking process

Benchmarking between the two selected function modeling approaches (i.e., function structures and IFM framework) is based on specific models of the same reference product,



Fig. 3. The glue gun.

Table 2. Criteria for function modeling benchmarking after Summers et al. (2013)

Representation Characteristics	
Characteristic	Explanation
Scope	Domain for which the function modeling approach is intended
Flexibility	Ability to modify and adapt the representation to address new problems
Indexing	Support access to the right (or useful) knowledge when needed
Consistency	Enforce physics and other consistency
Translationability	Tied to other engineering models
Behavior	Ability of the representation to simulation behavior
Scalability	Support both simple and complex problem types
Modeling Characteristics	
Characteristic	Question
Computational support	Is the modeling supported by software tools, restricted to human effort, or a mixed approach possible?
Construction approach	Are different construction types supported [i.e., forward chaining (moving from input to output)/backward chaining (moving from output to input)/nucleation/environment to system (outside to inside)/system to environment (inside to outside)]?
Decomposition and recomposition	Does the approach support decomposition and recomposition across multiple hierarchical levels and abstractions?
Cognitive Dimension Characteristics	
Characteristic	Question
Abstraction gradient	What are the minimum and maximum levels of abstraction? Can partial be created?
Closeness of mapping	What modeling conventions need to be learned? How intuitive is the resulting model?
Error proneness	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?
Premature commitment	Do the models require decisions before the needed information is available?
Secondary notation	Can the models be annotated or linked to other product representations?
Viscosity	How much effort is required to perform a single change? How easy is it to adapt the model from a model of a similar product?
Visibility	How easy is it to see all aspects of the model? Can two models be compared?
Reasoning Characteristics	
Characteristic	Question
Interpretability	How consistent and precise is the interpretation of the function models across different individuals, domain, and expertise?
Physics maintenance	Can questions about conversion of energy, material, irreversibility, or other physics be answered?
Analogical mapping	Does the representation support analogical mapping and alignment?
Pattern learning	Does the representation support learning of abstractions needed for analogical transfer?
State transformations	Does the representation support answering questions about different states?
Change propagation	Does the representation support discovery about the effects of perturbations in the system?

namely, the glue gun. The respective models were created in a collaborative effort by the authors. Initially, one author created the models alone. Subsequently, the models were evaluated by the second author, discussed, and gradually refined until no further changes were required. As the authors have comprehensive experience with the use of function structures and developed the IFM framework, prior training in the use of the approaches was not necessary. All function structures were created using a tablet computer enabling freehand sketches, which was almost identical to creating a sketch on paper. The models based on the IFM framework were created using standard office spreadsheet software and a basic process modeling tool.

The models were subsequently evaluated individually by both authors based on the benchmarking criteria listed in [Tables 2](#)

and [3](#). The initial evaluation results were discussed until consensus was reached wherever necessary. Based on the individual evaluations, observations during modeling, and reflection on the modeling process, the function modeling approaches were compared regarding their particular strengths and weaknesses.

5. MODELING THE GLUE GUN

The reverse engineering approach taken for this study allowed for functional analysis of the existing product (the glue gun). As a first step preceding the creation of the function models, the glue gun was completely disassembled (see [Fig. 3](#)). The resulting single components and distinct assemblies were uniquely named to allow referencing and discussion among

Table 3. Central function modeling perspectives after Eisenbart et al. (2013a)

Perspectives	Description
States	Representation of the states a system can be in or of the states of operands before (input) and after (output) a transformation process <i>Example:</i> the <i>state</i> of glue is changed from solid to molten by the glue gun
Transformation processes	Representation of the processes executed by the function carriers (technical products, stakeholders, etc.) that, from the designers' perspective, are part of the system under development and that may or may not result in a change in the state of the system or of operands. Therein, technical processes are transformation processes executed by technical systems (technical products, devices, etc.), whereas human processes are executed by stakeholders involved in function fulfillment (this explicitly includes human activities, e.g., during service execution) <i>Example:</i> heating the glue to <i>transform</i> its state
Interaction processes	Representation of interaction processes of stakeholders or of other technical systems, which, from the designers' perspective, are not part of a system, with stakeholders or technical systems, which are part of the system under consideration <i>Example:</i> the user interacts with the glue gun by <i>moving the trigger</i>
Effects	Representation of the required physiochemical effects, which have to be provided to enable, respectively support, the transformation processes that change the state(s) of operands and/or of the system into (a) new state(s) <i>Example:</i> friction is used to <i>transfer</i> the actuation force from the trigger to the glue stick
Use cases	Representation of different scenarios of applying the technical system for a specific purpose (e.g. fulfilling a goal, changing the state of the system or user); this is typically associated with the interaction of stakeholders or another technical system with the technical system under development (interaction processes), which triggers, respectively requires, subsequent processes to be carried out by the system <i>Example:</i> using the glue gun to connect parts, cleaning the glue gun after use, etc.
Technical system allocation	Representation of the role of technical products, their subsystems, or any other kinds of (tangible or intangible) technical means acting as function carriers in performing or enabling one or more functions; these technical means may be either part of the system under consideration or interact with it <i>Example:</i> the transformation process of "heating the glue" is performed by the <i>heating unit</i>
Stakeholder allocation	Representation of the roles of different stakeholders (humans or other animate beings), which may be users benefitting from a system or function carriers contributing to the system, e.g. through executing required processes or providing resources <i>Example:</i> the interaction process "moving the trigger" is performed by the <i>user</i>

the authors. The disassembly was necessary to understand the interactions of assemblies and parts, thus to understand the flow of operands, that is, materials, energy, and signals. The function structure was created first, followed by the creation of the IF model of the glue gun using relevant parts of the IFM framework. Both resulting models and the associated modeling processes are represented in the following subsections.

5.1. Modeling the glue gun using a function structure

Modeling of the function structure followed the original propositions by Pahl et al. (2007) and, hence, started by identifying the overall function of the glue gun and articulating a solution-neutral formulation of it. The purpose of the modeling activity was presumed to serve a reverse engineering project with the goal to analyze system functionality and to identify improvement potentials. Therefore, a decomposition of the overall function and a simultaneous analysis of assumptions that constrain the solution space were conducted. For every decomposition and detailing of the model, all assumptions, respectively, decisions that constrain the solution space, were documented complementary to the respective model. The initial overall function was decomposed three times and detailed multiple times. In total, five different models were subsequently created. The final model was then reworked for a better visual appearance (see Fig. 4).

The solution-neutral formulation of the overall function of the glue gun is "connect parts" (see Fig. 5, left-hand side). The desired outcome of the use of the glue guns are "connected parts" and the input are "unconnected parts."

An important assumption about the product is the used connection principle. The glue gun connects parts by bonding. Alternatives (on the same level of detail) would be form fit or force fit connections. An advantage of bonding is the absence of additional parts or connecting elements. As gluing, or adhesive bonding respectively, was chosen to connect parts instead of brazing and welding, further assumptions/constraints for the product are that the parts might not be allowed to be exposed to high temperatures, which go hand-in-hand with these processes. An alternative assumption would be that parts are not metallic, thus limiting applicable bonding techniques. Therefore, the overall function was reformulated into "glue parts" (see Fig. 5, right-hand side).

The next decomposition steps of the overall function entailed creating lists of subfunctions and auxiliary functions, and related underlying assumptions constraining the solution in such a way that the design converges into the studied glue gun (see Table 4). In the first step, it became clear that whatever glue is used the flow of material will require storing, application, moving, and dosing of the material (see Table 4, column describing third model). One assumption made at this point that affected the modeling activity noticeably is the definition of the system boundary, that is, excluding the

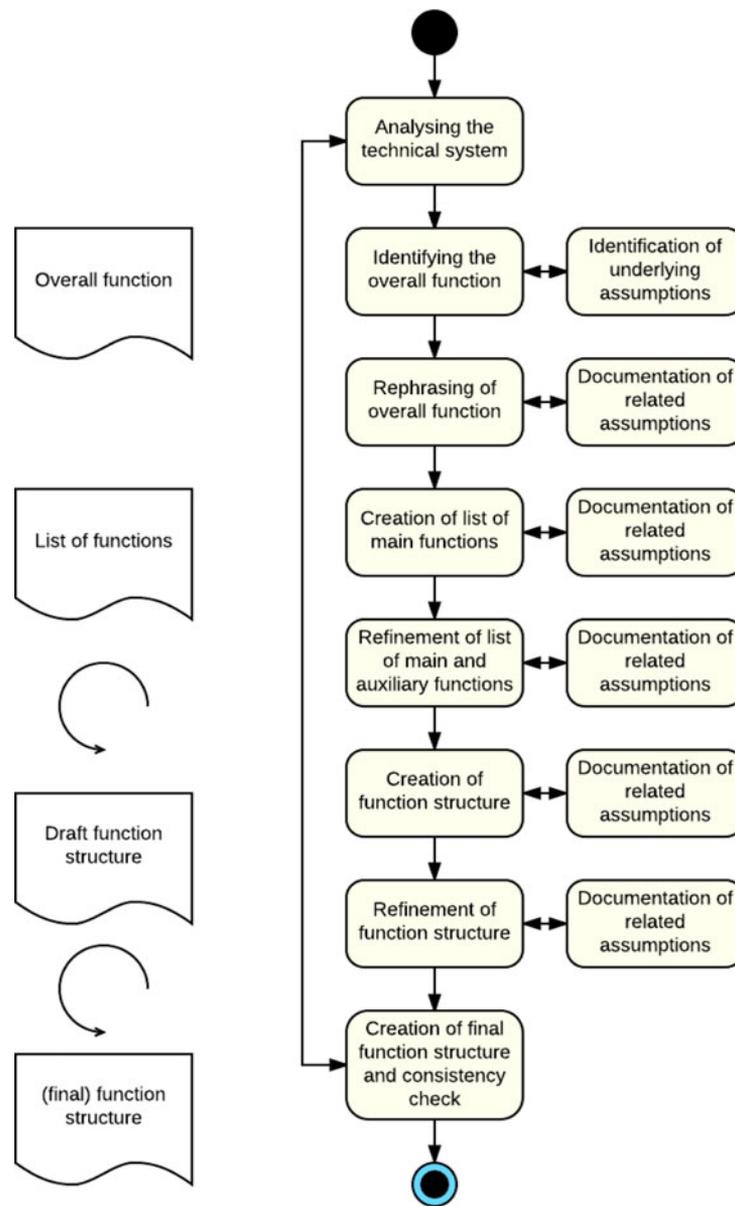


Fig. 4. Modeling process of the function structure.

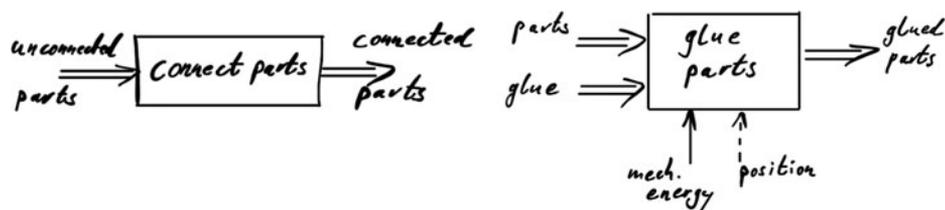


Fig. 5. Overall function (left) first version and (right) second version.

user from the modeled system; thus, users are treated as external to the functional system that constitutes the glue gun.

Probably the most important assumption is the choice of the glue, for this example the use of a thermoplastic adhesive

formed as a cylindrical stick. The use of this type of glue requires thermal energy to change the state of matter from solid to liquid. Thus, several additional functions are required enabling the gluing of parts (see Table 4, column describing fourth model).

Table 4. List of functions contributing to overall function “glue parts”

	First List (Third Model)	Second List (Fourth Model)
Underlying assumptions	Focus on technical system (some functions may need to be carried out by involving users, i.e., related to the interface to the user)	Type of glue Hot glue Glue sticks Need for thermal energy
Main functions	Store glue Apply glue Move glue Dose glue	Hold glue stick Move glue stick Move glue stick Heat glue Store melted glue Apply melted glue Move melted glue Dose melted glue
Auxiliary functions	Move device	Channel heat Channel heat from source to glue Protect user Protect device Connect/disconnect with/from energy source

The third and the fourth models show the required functions for the overall function “glue parts” on different levels of details incorporating the assumptions made during each decomposition step. Both models are pure lists of main functions, auxiliary functions, and underlying assumptions that constrain the design. In the next step, the interdependencies (i.e., the flows of matter, energy, and signals) were added, resulting in a function structure. The function structure was modeled starting with what we consider the main operand flow by virtue of the overall function, that is, the flow of glue (material). The remaining flows and activities were modeled succeeding this step. The function structure was discussed between the authors and gradually refined afterward.

The resulting function structure is shown in Figure 6. At this stage, the authors decided that further detailing, while being possible, was not expected to provide added benefit. Further detailing would make sense in case a rather specific analysis of the product, the transformation processes, or design alternatives was required by the designers, yet not for the purpose of the comparison shown here.

5.2. Modeling the glue gun using the IFM framework

For modeling the glue gun, not all of the views/contents in the IFM framework are required. As discussed earlier, depending on demand, only a selection of views may be needed depend-

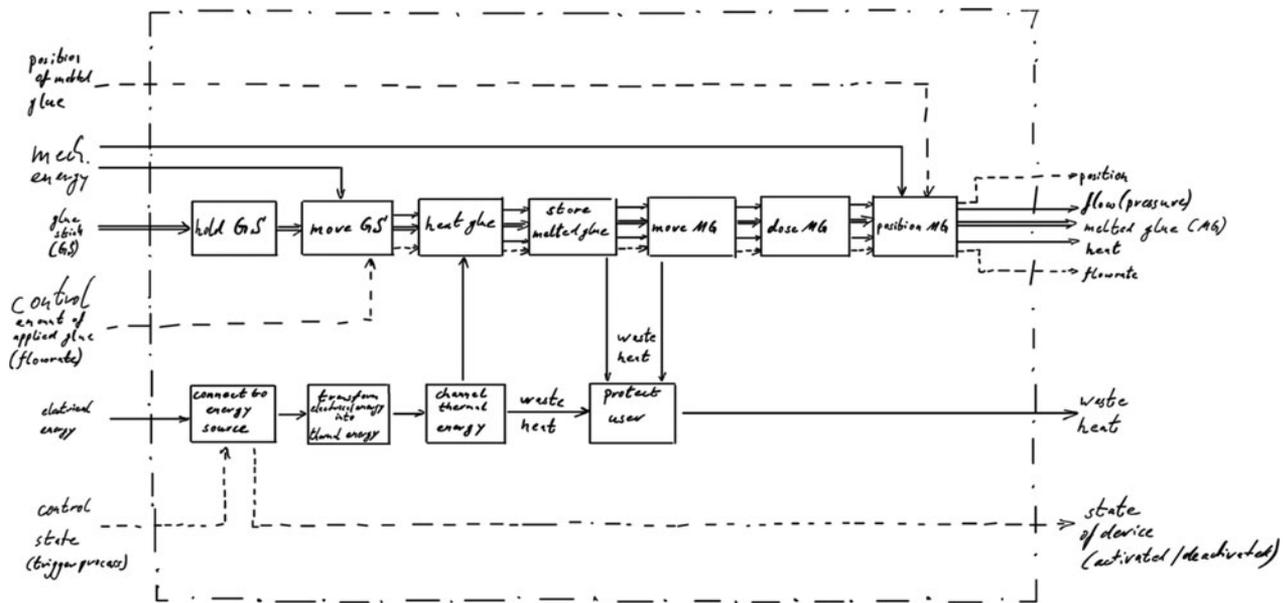


Fig. 6. (Final) function structure.

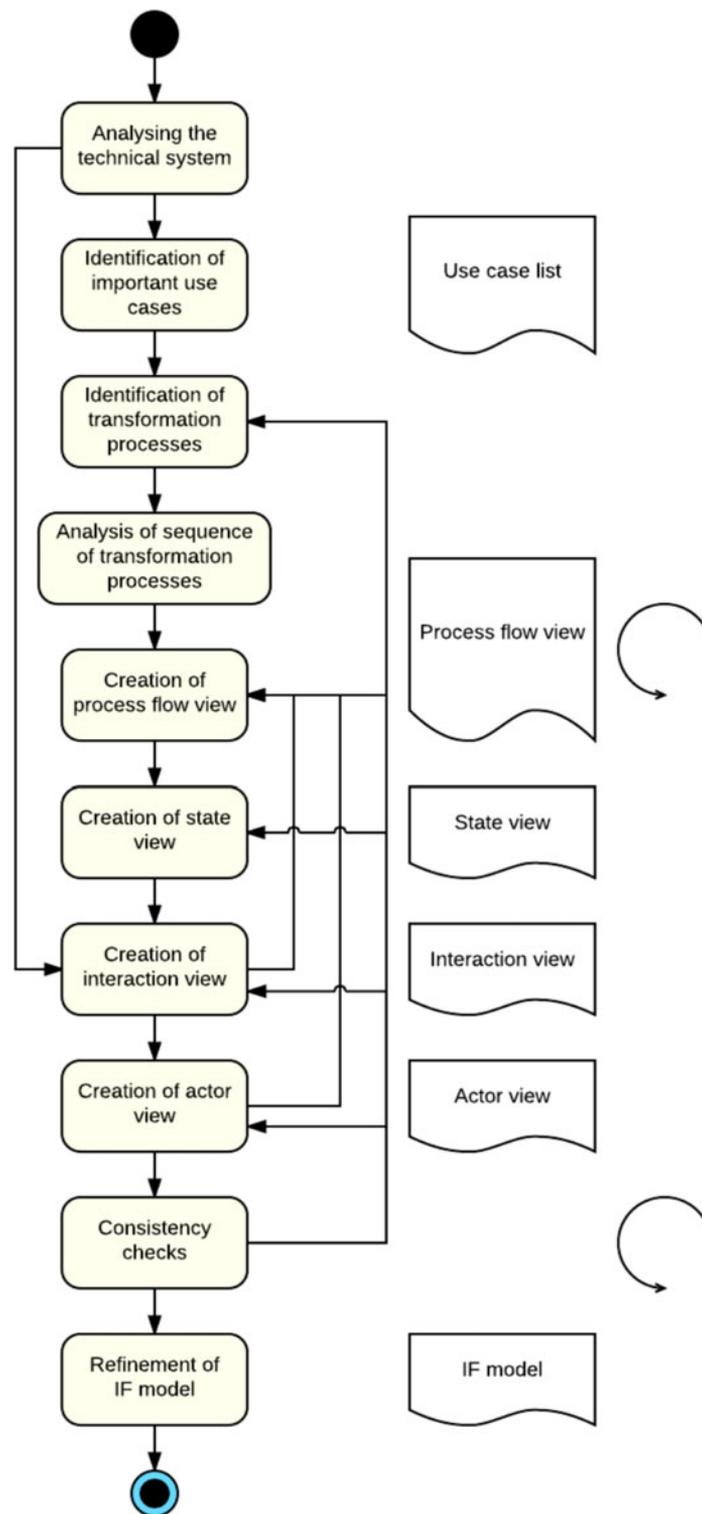


Fig. 7. Modeling process of the integrated function model.

ing on the specific purpose of modeling. For the example of the glue gun, the *process flow view*, *actor view*, *state view*, and *interaction view* were selected.

In accordance with recommendations by Eisenbart et al. (2016a) and Andreasen et al. (2015), modeling the glue

gun with the IF model (the specific model based on the IFM framework) was set up starting with an identification of central use cases (see Fig. 7). Central use cases comprise the following: “prepare” (unpacking the glue gun and loading a glue stick), “use” (using the glue gun for gluing parts),

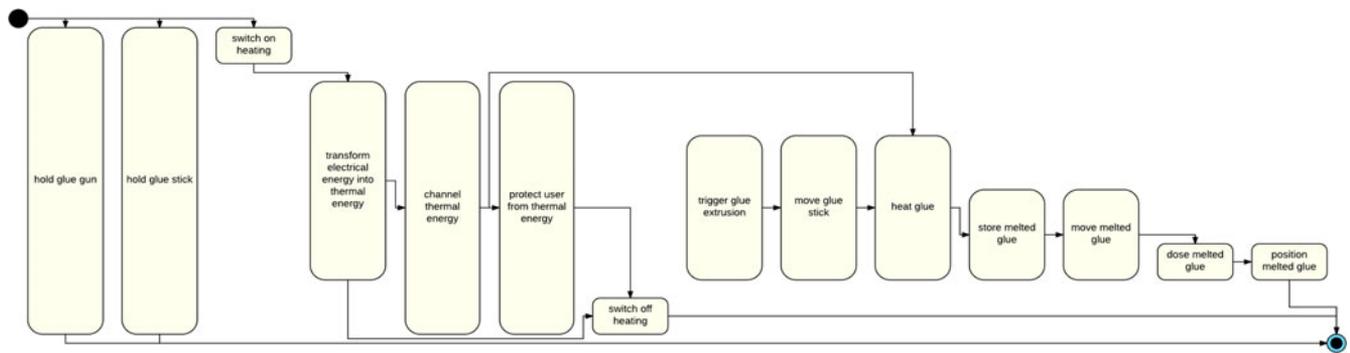


Fig. 8. Process flow view for the glue gun.

“maintain” (cleaning the glue gun and checking for damages), and “repair” (disassemble and repair). The list of use cases constitutes the *use case view*.

The model was detailed by moving from determining the central use cases envisaged to the specific interaction and transformation processes required within them. The process flow view represents the sequence of interaction and transformation processes that constitute a single use case, that is, the interdependencies between individual processes (connectors) and the temporal order (vertical axis) of these processes. For the comparison shown here, the remaining views of the IF model were created only for the use case of “use,” as the most relevant in accordance with the overall function fundamental to the function structure modeled earlier and thus critical to this comparison. Afterward, the *process flow view* (see Fig. 8) was generated by gradually defining the processes that constitute the selected use case by determining associated flows of operands and the sequence of processes, as well as the states of related operands (*state view*) until these were logically consistent and complete.

For the purpose of reverse engineering, we analyzed the system architecture and mutual impacts of parts in the glue gun, thus constituting the *interaction view* (see Fig. 9), by adding actors and their interdependencies. Each interaction between actors and operands is indicated with a “1.” The type of interaction is not specified in this model, but could be detailed if required.

We linked the *interaction view* (see Fig. 10) to the *process flow view* using the *actor view* and the *state view* (see Fig. 11). While the creation of the different views was done more or less sequentially, the refinement and check for consistency of the modeled views and (in extension) of the entire IF model was incorporated using the integrated model combining all generated views as shown in Figure 12.

5.2.1. Adaptation of the IFM framework

The creation of the IF model illustrating the glue gun was facilitated by slight structural adaptations based on the original IFM framework. Adaptation was done along the modeling process to match the needs of this particular modeling activity (see Fig. 13). This encompasses a simplification of the *use case*

view as only one use case was modeled in more detail in the presented study. The *interaction view* has been placed on the right-hand side, solely because this simplified the manipulation of the model in the used software tool operatively. At the same time, the actors and operands placement in the matrix was swapped as this allows an easier identification of user and operand involvement in the individual processes, thus facilitating the readability of the *actor view*. The *actor view* has been expanded to include actors and operands, as involvement of operands in processes was important for the analyzed system. The *state view* has been reduced by limiting the indication of states to operands and omitting indication of states of actors as the temperature and consistency of the glue is the most relevant state change. These adaptations were perceived useful, as the indication of operands involvement in the *actor view* seemed necessary for checking consistency of the *process flow view* while the modeling of state changes of actors in the *state view* seemed not necessary for this simple example.

6. COMPARISON AND DISCUSSION OF THE MODELING APPROACHES

The needs of practitioners differ depending on the individual context. Thus, the question remains which modeling approach is best suited to supporting design activities in a particular context. Function modeling approaches can be compared on the basis of numerous characteristics that have a profound influence on their suitability (see Tables 2, 3). Some factors that stand out will be discussed in the following sections.

The discussion comprises two levels (wherever possible). For each main theme, the potential of the function modeling approaches is discussed more generally. Wherever possible, the differences found in the specific generated models of the glue gun, that is, the function structure and the IF model (see Figs. 6, 12), will be discussed in more detail and set into relation to the general discussion. In order to distinguish clearly between modeling approaches and specific models of the glue gun, the approaches are referred to as the IFM framework and function structures (plural), whereas specific models are referred to as the IF model and function structure (singular).

	User	Operands	Glue	Electrical energy	Thermal energy	Mechanical energy	Heater assembly	Heating unit	melting chamber	heating cartridge	foil	Cable unit	cable	connector	insulation	cable guidance	Trigger assembly	Trigger unit	Trigger	Rod	Lever	Spring	Slider unit	Slider	Pin	Case assembly	Case unit	right case	left case	screws	Thermal insulation unit	Nozzle insulation	Chamber insulation	Ring	Spring			
User	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	
Operands	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
Glue	0	1	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	1	0	0	
Electrical energy	0	1	0	0	1	0	0	1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Thermal energy	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	
Mechanical energy	0	1	1	0	0	0	0	1	1	0	0	0	1	1	0	1	0	1	1	1	1	1	1	1	1	0	0	1	1	1	1	0	0	0	0	0	1	
Heater assembly	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	
Heating unit	0	1	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	
melting chamber	0	1	1	0	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	
heating cartridge	0	1	0	1	1	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
foil	0	1	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Cable unit	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
cable	0	1	0	1	0	1	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	
connector	0	1	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
insulation	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
cable guidance	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	
Trigger assembly	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Trigger unit	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Trigger	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
Rod	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Lever	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Spring	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Slider unit	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Slider	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0
Pin	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Case assembly	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Case unit	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
right case	1	1	1	0	1	1	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	1	1	0	1	1	1	1	1	1	1
left case	1	1	1	0	1	1	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	1	0	0	1	0	1	1	0	1	1	1	1	1	1
screws	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
Thermal insulation unit	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Nozzle insulation	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
Chamber insulation	0	1	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0
Ring	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0
Spring	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0

Fig. 9. Interaction view.

Both parts of the discussion contribute to a more holistic comparison of both function modeling approaches. Findings that relate to the modeling approaches are summarized in the tables that relate to the individual sections of the discussion.

6.1. Scope and content

The perspectives taken for function reasoning and function modeling differ between design disciplines (Eisenbart et al., 2012). The type of content addressed in function models varies across disciplines, and therefore most function modeling approaches are only used in one or few particular disciplines. Function structures, most prominently representing transformation processes, primarily meet the needs of mechanical engineers (which is not to say others cannot use them). The IFM framework, in contrast, was designed to integrate contents

found in function models across a broad range of (engineering) disciplines and can thus be used in an interdisciplinary context (including mechanical engineering, software design, service design, mechatronics, and PSS design), where it is imperative for designers to overcome limitations of individual disciplines (see Table 5).

In the IF model (see Fig. 12), the vertical direction in the process flow view (see Fig. 8) illustrates the causal flow between transformation processes. The sequence and parallelism between different transformation processes and effects is derived from the chronological succession, in which operands are transformed from system input to output. The concept of causal flows in the process flow view similarly applies to the function structure (see Fig. 6), though here it concerns the flow of operands and changes in aspects other than the temporal sequences.

hold glue gun	hold glue stick	switch on heating	transform energy	channel thermal energy	protect user	switch off heating	trigger glue extrusion	move glue stick	heat glue	store melted glue	move melted glue	dose melted glue	position melted glue	User
X		X				X	X	X			X	X	X	User
				O	X	X	X	X	X	X	X	X	X	Operands
								O		X	X	X	X	Glue
		X	X	X	X	X			X					Electrical energy
			X						X					Thermal energy
X					X		X	X			X	X	X	Mechanical energy
	X	X		X		X			X	X	X	X		Heater assembly
	X	O		X		O			X	X	X	X		Heating unit
		X		X		X			X	X	X	X		melting chamber
		O	X	X		O			X					heating cartridge
				X					O					foil
		X				X			O					Cable unit
		X				X			O					cable
		X				X			O					connector
		O				O			O					insulation
							X	X			X			cable guidance
							X	X			X			Trigger assembly
							X	X			X			Trigger unit
							X	X			X			Trigger
							O	X			X			Rod
							O	X			X			Lever
							O	X			X			Spring
	X						O	X			X			Slider unit
	X						O	X			X			Slider
							O	X			X			Pin
O	X				X		X	O			O		X	Case assembly
O	X				X		X	O			O		X	Case unit
O	X				X		X	O			O		X	right case
	X				X		X	O			O		X	left case
					O		O	O			O		O	screws
				X	X				O					Thermal insulation unit
				X	X				O					Nozzle insulation
				X	X				O					Chamber insulation
	X			X	X				O					Ring
				X	X				O					Spring

Fig. 10. Expanded actor view.

initial states		solid, cold	not flowing	void	void
			switch on heating		supporting "switch on"
states			flowing	void	dissipated
states		solid, cold	supporting "transform electrical ...energy"	"transform electrical ...energy"	void
states		"heat glue" till "position melted glue", ...			supporting "trigger glue extrusion",
states		"position melted glue"			supporting "move melted glue"
states		melted, hot	flowing	dissipated	dissipated
		switch off heating			supporting "switch off"
final states		solid, cold	not flowing	dissipated	dissipated
	User				
	Operands				
	Glue				
	Electrical energy				
	Thermal energy				
	Mechanical energy				

Fig. 11. Reduced state view.

The IFM framework was developed with the goal of incorporating all modeling perspectives presented in Table 3; thus, it is no surprise that the framework covers all the perspectives listed in Table 5 (lower part of the table). However, the list of

modeling perspectives is included in this comparison to more clearly highlight the differences between both approaches.

In function structures, transformation processes and effects are generally not separated, but are interconnected and modeled

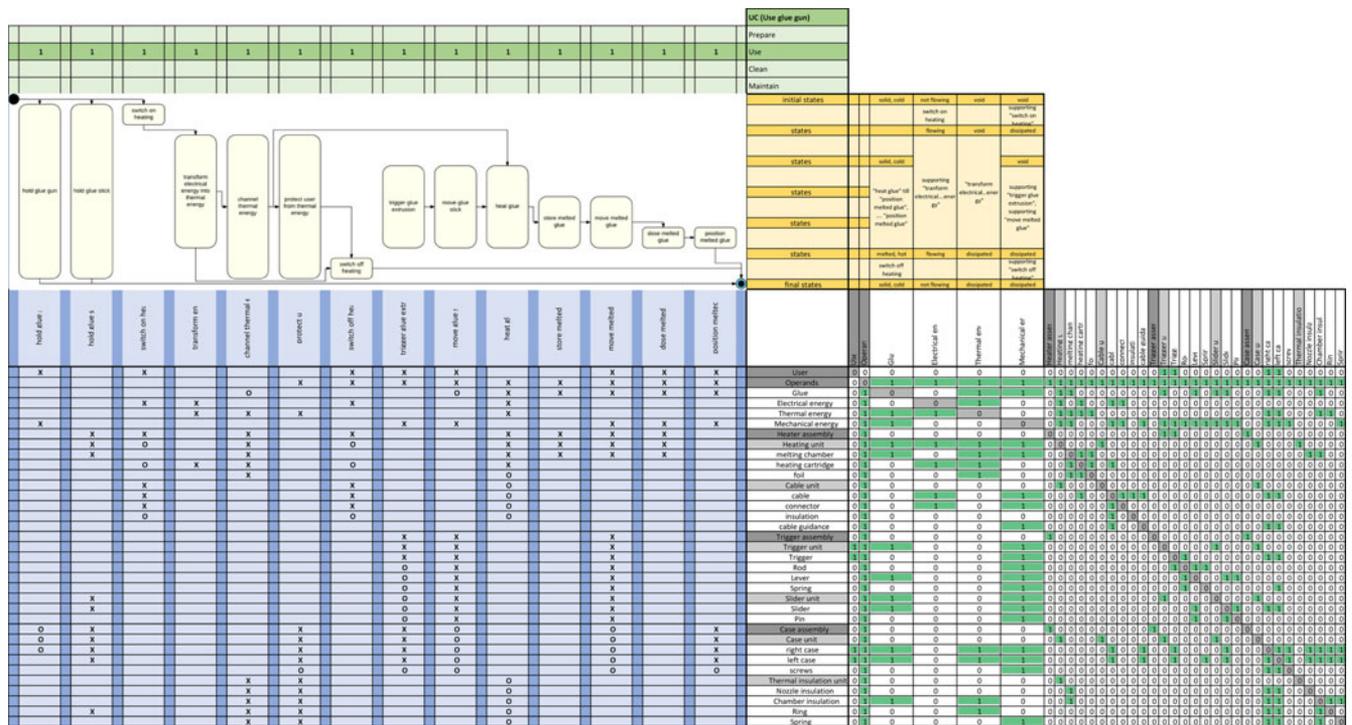


Fig. 12. Integrated function model of the glue gun.

in the same representation. The IFM framework allows linking additional views such as the *effect view* to support detailed analyses of specific transformation processes; this was not carried out in the IF model of the glue gun here, but can be added seamlessly in a concrete design project, if required.

6.2. Adaptability and reasoning support

Function structures do not inherently allow for substantial flexibility, that is, modification and adaptation. The literature

explicitly recommends the rigorous use of the proposed notations to help users interpret models more easily and consistently. In line with this, the created function structure is comprehensive, as per its original conceptualization (see Pahl et al., 2007).

The modular and interlinked setup of the IFM framework supports analogue mapping between the different views, and thus between different entities contributing to function fulfillment. Depending on the modeling task, the use of the different views can flexibly be adapted to the specific demands. As only simple modeling notations are used for creating the matrices

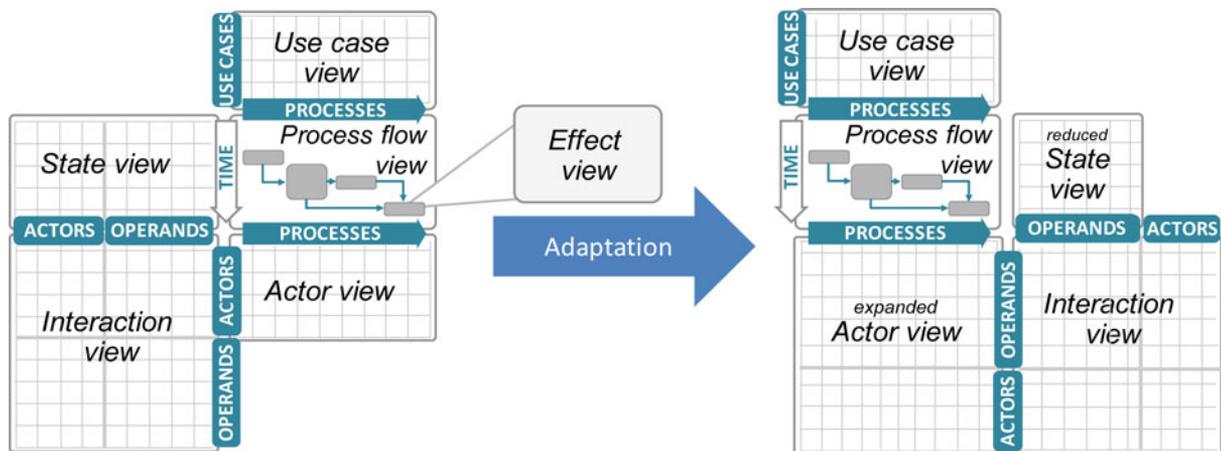


Fig. 13. Adaptation of the integrated function modeling framework.

Table 5. Comparison of scope and content

Characteristic	Function Structures	IFM Framework
Scope	Primarily for mechanical engineering	By virtue of its cross-disciplinary setup, the IFM is meant to support interdisciplinary function modeling, including mechanical engineering, software, service, mechatronic, and PSS design.
Behavior	Simulation of behavior is not possible.	Alternative flows pertaining to behavior are made explicit (see Eisenbart, Mendel, et al., 2015) but not seamlessly simulated.
Physics maintenance	Questions about conversion of flows of operands can be answered.	Questions about conversion of flows of operands can be answered.
State transformations	Function structures do not per se inform about states of operands, but this information can be added to the flow of operands.	Each state change can be specified in the <i>state view</i> of the IFM.
Perspectives	Function structures	IFM framework
States	Can be modeled (see example glue stick vs. melted glue)	States of operands and actors are indicated in the <i>state view</i> .
Transformation processes	Is modeled with clear distinction of flows of matter, energy, and signals	Transformation processes are modeled in the <i>process flow view</i> .
Interaction processes	Interactions are not explicitly modeled. Incoming and outgoing flows are unspecific because the corresponding actor is not specified.	Interactions are indicated in the <i>interaction view</i> .
Effects	Effects are not modeled.	Effects can be modeled and linked to (or embedded) in the <i>process flow view</i> .
Use cases	Use cases are not modeled or specified.	Use cases are distinguished in the IFM, individually modeled, and involvement of processes in use cases indicated in the <i>use case view</i> .
Technical system allocation	Technical system allocation is not modeled.	The interdependencies and involvement of technical systems into function fulfillment is modeled in the <i>actor view</i> .
Stakeholder allocation	Stakeholder allocation is not modeled.	The interdependencies and involvement of stakeholders into function fulfillment is modeled in the <i>actor view</i> .

and flowcharts, interpretation of the content is seamless. Even following adaptation of the IFM framework's structure (see Figs. 12, 13) and morphology, the model supports consistent interpretation by different model users.

The consistent interpretation of function structures depends on the experience of modelers and model users and on the consistent use of the proposed notations during model building. The proposed notations for function structures are per se not flexible, and thus should enable consistent interpretation; however, the morphology of the model is not regulated, which makes it often difficult to derive a consistent interpretation (see Table 6). In the created IF model of the glue gun, we made several slight adaptations to improve its applicability to the given modeling task. The authors are used to working with the model, yet it is aspired that novices adopting the model would be able to learn how to use it rather quickly as well, given that the model is entirely based on well-known matrices and simple flow modeling.

6.3. Visualization and manipulation capabilities

The separation of the modeled entities into different views in the IFM framework and indication of interdependencies facilitates the search for flaws and missing information and eases the gradual development of a model, as shown for the glue gun example (see Fig. 7). When modeling, or engaging in function analysis for an existing system, the setup of inter-

linked matrices visually facilitates logical consistency analysis across views. This can be achieved through deductive logic and continuous comparison of whether the contents in each view are inherently consistent. For instance, in the case of the "heat glue" process, it can be concluded that there must be an associated operand "glue" whose temperature in the input state is lower than in the output state.

Function structures are less rigidly structured, which would help designers to build the model, and thus designers have to scrutinize model content and model structure conjointly. The IFM framework in turn allows designers to focus on modeled content more easily.

Such reasoning supports gradually entering information into the views of the IFM framework concurrently during model generation. It can be similarly applied during change management, for instance, to ensure that the model maintains its logical consistency. Compared to function structures, this facilitates model manipulation, which is supported by the use of matrices for indicating interdependencies, which can be manipulated and compared easily, compared to a purely graphical process model, and indication of interdependencies as done by function structures (see Table 7).

6.4. Interconnectedness

Function modeling has been used in multiple stages of the design process (product planning and task clarification,

Table 6. Comparison of adaptability and reasoning support

Characteristic	Function Structures	IFM Framework
Flexibility	Flexibility of the approach is very low and rigorous application is explicitly demanded in the literature.	Flexibility of the modeling approach is one of the core attributes of the IFM framework, as evidenced here by the implemented changes constituting the final IF model to optimally fit the modeling task at hand.
Interpretability	Consistency of interpretations of a model among individuals that contributed to the creation of a specific model is good, even though diversity increases with the passing of time. Consistency of interpretation requires experience in creating and reading function structures and depends on the quality of the model.	Because our own impression might be biased, we rely on feedback obtained from practicing designers using the framework. Interpretability of the modeled information was never mentioned as a problem; the framework has been praised for its ease of use (see Eisenbart et al., 2016a).
Closeness of mapping	A simple and commonly used notation for modeling systems is used. The three main flows (material, energy, and signal) are differentiated using a specific notation.	The IFM requires basic understanding of DSM modeling and flowchart modeling. Aside from the existence of the different views, no specific notation is used.
Analogical mapping	Analogical mapping is not supported by the modeling approach.	The integrated function modeling framework requires analogical mapping. The modular and interlinked setup of matrices representing interdependent domains is key to the modeling approach.
Pattern learning	The approach does not support learning of abstractions needed for analogical transfer.	The approach presumes basic skills in analogical reasoning. Using the IFM will train analogical reasoning skills, because interpretations of the relational structure of elements can be verified.

Table 7. Comparison of visualization and manipulation capabilities

Characteristic	Function Structures	IFM Framework
Indexing	Clear indication of different flows allows easy tracking. Because diversity of entities is limited, accessibility to modeled information is good.	Although many different entities are included in the model, specific information can be traced easily because entities are allocated in particular views and interdependencies between them are clearly indicated.
Consistency	If applied properly, flows of operands and transformations thereof can be consistently modeled.	Separation of entities into different views and indication within and among groups of entities enforce consistency of the model (if modeled to full extent).
Error-proneness	A common mistake when creating function structures is that function descriptions diverge from the verb–noun notation by using verbs only or referring to specific components or subsystems.	A possible failure source is to oversee interdependencies; to date, no software support is available that would check for such mistakes.
Hidden dependencies	Dependencies need to be indicated manually by modeling the different flows.	Dependencies need to be added manually and are clearly indicated. The IFM distinguishes between different types of interdependencies between views and among the modeled entities.
Viscosity	Changes can easily be introduced but need to be reworked substantially (because the structures are usually created manually).	Changes can be easily introduced. The organization of the IFM allows easy consistency checks of added or changed information. The reuse of a model is easy thanks to the documentation of interdependencies. Reuse of large parts of the IFM is necessary when modeling multiple use cases.
Change propagation	Perturbations caused by changes in the order of modeled functions can be identified. Changes in other domains representing the system under consideration cannot be analyzed.	Perturbations of the system can be identified and change effects across the modeled views (entities) can be traced and analyzed easily.

conceptual design, and embodiment design) for different purposes (analysis of the problem, synthesis of the solution, and documentation; Gericke, Qureshi, et al., 2013, Eisenbart et al., 2016b). Hence, the potential benefits yielded by a

function model depend to some extent on its ability to be embedded in a meaningful way in the design process, that is, to link function modeling to other design activities (to link the contents to other engineering models/documents)

and to use the modeled information in multiple design steps/stages.

The functions modeled in function structures serve as a starting point for the subsequent search for working principles enabling respective functions. Mapping of functions and matching alternative working principles can be supported using morphological charts. A consistency analysis of combined working principles based on the flow of operands as modeled in the function structure may afterward help to identify compatible working principles that can be integrated into a working structure, that is, a solution concept (see Pahl et al., 2007).

The interconnectedness of the IFM framework to other engineering models goes beyond this endeavor (see Table 8). It enables adaptation, including adding complementary views, representing additional information such as cost information and risk assessment data. The “open” interface to other models, which can be linked to one of the views via the included entities, enables the application of established analysis methods for dependency structure matrices (see Lindemann et al., 2009; Eppinger & Browning, 2012) to system functionality. These include (but are not limited to; see Eisenbart, Gericke, et al., 2015) the following:

- conflict analysis between (mutually dependent/exclusive) entities;
- change prediction and effect analysis concerning elaboration on the effects of implementing changes to operands and actors, provided functionality, use case fulfilment, and so on;
- evaluating optimization potentials such as modularization opportunities or comparative analysis of solution variants;
- criticality and failure effect analysis of functions and actors from system level down to individual elements and their interactions; and
- cost analysis and prediction in change management.

The main advantage of the IF model (see Fig. 12) over the function structure (see Fig. 6) for the presented example is the integration of product architecture information (*interaction view*, Fig. 9) via the *actor view* (Fig. 10) and *state view* (Fig. 11). This integration of product architecture information

allows the linking of many analyses in later design phases to the functional description of the product (see Eisenbart et al., 2016a).

The different models of the glue gun (see Figs. 6, 12) also show that linking of function models is as important as linking function models to other engineering models. It is important as the modeled content may originate from different disciplines, which prefer to use specific modeling approaches before integrating that information into other models. An example of how the IFM framework enables a discipline-specific model is shown by the possible evolution of the IF model's *process flow view* (see Fig. 8) out of the function structure (see Fig. 6). Furthermore, it is important as knowledge gain during function modeling itself is a gradual process, and thus information is complemented stepwise. Function structures allow such gradual information accumulation as shown for the glue gun example, but they are limited compared to the IFM framework, as only few perspectives (entities) are included (see Table 5).

6.5. Modeling characteristics

Both modeling approaches allow detailing (decomposing) certain parts of the respective function models, while other parts are modeled at an abstract level (Table 9). The evolution of the function structure starting with the overall function (see Fig. 5) of the glue gun and its stepwise decomposition that results in the function structure shown in Figure 6 illustrate this very well. A similar approach can also be applied for creating an IF model with more or less detail than depicted in Figure 12. However, in this article, only one version of the IF model is shown.

The available tool support for modeling function structures is limited to simple graphical modeling tools or paper-based sketches, making editing a cumbersome activity. The use of spreadsheet software for modeling with the IFM framework allows gradual detailing of parts of the model while keeping other parts more abstract. This reduces the effort for refining and changing the model and hence contributes to the usability of the modeling approach. The effort for creating the IF model shown in Figure 12 compared to the effort for creating the function structure shown in Figure 6 was around four times higher. This is hardly surprising seeing that the IF model

Table 8. Comparison of interconnectedness

Characteristic	Function Structures	IFM Framework
Translationability	It is difficult. An example is the subsequent use of identified functions in morphological matrices for solution finding and combination of working principles (see Pahl et al., 2007).	It is possible, but not built into the model. Potential is discussed in Eisenbart, Dohr, et al. (2013). Prototypical linking of IF modeling with requirements lists and simulation models has been achieved using a software tool chain with partly manual data exchange (see Dohr et al., 2014)
Secondary notation	A function structure is usually not linked to other product representations or annotated.	Annotation of the IFM is possible. Linking to other product representations such as requirements lists or BIMs is possible.

contains much more information about the analyzed system, the glue gun, and provides more opportunities for later use during the design process. From our perspective, this justifies the extra effort if the designers intend to use it later. Conversely, if function modeling is mainly used as a means for documentation or an early conversation starter in a design step process, such detail may not be required and function structures might be better suited. Yet, in the concrete benchmark problem addressed here, that is, reverse engineering of the glue gun, the IF model provides additional benefits, as the link between the system's functionality and its product architecture is fully defined. This support for comprehensive analysis should inherently help in making design changes more plannable.

The readability of function structures depends mainly on the number of modeled flows and the organization of the model. More experienced modelers will manage to make a function structure easy to read. As function structures are gradually built, the organization of the modeled elements is often suboptimal in the first version, and thus it is necessary to revise the model to optimize readability, even though this requires extra time. We did the same with the function structure of the glue gun.

In comparison, the complexity and readability of an IF model is determined to a large extent by the level of detail of the *interaction view*. The number of elements modeled in the *interaction view*, which basically represents the system architecture, affects the size of the matrix. In order to keep the complexity low and improve readability, modeling at a subsystem level might be sufficient for most modelling purposes. For the glue gun, which is not complex, the *interaction view* (see Fig. 9) is detailed, representing almost every single part. If required for specific purposes, the granularity of the model can be changed for particular subsystems. Modern spreadsheet software allows balancing the granularity of the model and its readability by allowing partial fading of the model, if desired.

6.6. Effects of premature commitments: Solution-neutral function modeling

Function modeling is a means to support and document the results of function reasoning. This helps designers to reflect on the modeled information, revise and refine content, and gradually complement the model by expanding and detailing it as required (Table 10). This is exemplified for both modeling approaches. The gradual detailing of a function model is shown based on the function structure in Figures 5 and 6 and Table 4 and for the evolving IF model in Figures 8–12.

Completely solution-neutral modeling of the functionality of a system is almost impossible. The model building process requires making assumptions that will constrain the solution space (see Fig. 4). The more detailed the model becomes, the more assumptions need to be made; thus, designers need to balance the level of detail of a model (its granularity) and the constraints for the solution space.

A problem with function modeling is that the documentation of such assumptions is usually not considered to comprise part of a function model, even though the assumptions affect the building and understanding of the model, which we experienced very clearly in setting up the function structure of the glue gun. A consequence of this is that the underlying assumptions often are neither made explicit nor documented, and thus this information is lost even though it is important for understanding the model in the way intended by the modeler. Identifying such assumptions is a crucial part of the reverse engineering process. A clear understanding of assumptions and constraints that are likely to have guided the original design process provides insights for possible design interventions during the redesign/modification of the analyzed system. To ease this process, the shown function structure (see Fig. 6) was created starting with the overall function fulfilled by the glue gun (see Fig. 5).

Whether unconscious assumptions/design decisions affect the design process depends greatly on the modeler (i.e., the designer), the selected function modeling approach, and how the model is used. If the model is understood as a sketch representing one of many alternatives, this will become less of a problem, as the designer (or team) needs to evaluate alternative function models and the convergence of the solution space is based on a comparison of alternatives. If the function model is understood as solution neutral in itself, thus neglecting the constraining effect of unconscious assumptions, the solution space is reduced and design alternatives are excluded in an (probably very often entirely) unconscious manner.

Function structures as described by Pahl et al. (2007) do not require the documentation of assumptions as shown in Figure 4 and included in Table 4. We documented said tacit knowledge (as per the discussed assumptions) for the glue gun example as a complementing analysis serving the goal of reverse engineering this product. Even though not prescribed in the literature, assumptions can of course be documented, and thus this limitation was overcome. However, it remains challenging to compare alternative function structures because the used modeling notations do not regulate the organization and modeling of flows.

The IF model, while also not being solution neutral, is expected to make the designer more conscious of the gradual convergence of the solution space by virtue of the saliency of allocated solution elements. This is, after all, the manifestation of the decisions made in relation to which components, subsystems, or human actors have been *selected* by the designers to fulfill the desired functionality.

6.7. Limitations of the comparison

The comparison is limited insofar as it considers only one particular function modeling task, namely, the reverse engineering of a glue gun, a low-complexity product. For a comprehensive comparison of the two selected function modeling approaches, other challenges embedded in the synthesis of original designs or evolutionary design of more complex

Table 9. Comparison of modeling characteristics

Characteristic	Function Structures	IFM Framework
Scalability	Simple and complex can be modeled.	Simple and complex systems can be modeled.
Computational support	Modeling can be done using sketching and drafting tools. To the best of our knowledge, there is no tool support for automated model creation or model analysis.	Although modeling can be done manually on a white board, this is only useful for simple or high-level models. More detailed or complex models need software tool support (spreadsheet software and process modeling software). Analysis of the matrices can be supported using spreadsheet software (because they can be treated as DSMS or DMMS; see Browning & Eppinger, 2012).
Construction approach	Different construction types are possible.	Different construction types are possible. A combination of construction types is useful to gradually elaborate the model and perform consistency check in parallel.
Decomposition and recomposition	The approach supports de- and recomposition across multiple hierarchical levels by creating individual models for each level.	The approach supports de- and recomposition across multiple hierarchical levels. <i>State view</i> , <i>actor view</i> , and <i>interaction view</i> can be modeled considering different levels in one model. Decomposition of processes in the <i>process flow view</i> is possible using containers for a subset of processes that form a higher-level process.
Abstraction gradient	The modeling approach can model a single use case at different levels of abstraction ranging from an overall function to physical effects that enable a single subfunction. Subfunctions can be decomposed individually (thus, partial modeling is possible).	The modeling approach can model multiple use cases at different levels of abstraction ranging from an overall function to physical effects that enable a single subfunction. Modeling of physical effects using the <i>effect view</i> can be done using alternative modeling approaches. Creation of partial models is possible in many ways (e.g., on different levels of abstraction and for individual subsets of entities and for selected subsystems).
Visibility	The model is easy to read. Comparison might be difficult as the organization of flows is not regulated.	Readability of the model depends on the complexity of the modeled system and the level of detail. Because matrices can become complex, capturing all information at once might be challenging, but all information is accessible at the same time. Modern spreadsheet software allows stepwise fading of subordinate hierarchical levels of the modeled system, and thus simplifies reading of and interacting with the model. Comparison of multiple models is possible because the models are organized in the same way. Detailed comparison can be done by using analysis capabilities of spreadsheet software.

systems need further attention. Furthermore, a metastudy of multiple benchmarks on different function modeling challenges would enable deriving strengths and weaknesses of further function modeling approaches for comparison purposes.

The benchmarking protocol proposed by Summers et al. (2013) does not provide metrics for the evaluation of the compiled characteristics. Therefore, the presented results are rela-

tive by nature and rely solely on our personal observations of their application in the past (in industry workshops and teaching) and on our personal reflection during modeling the glue gun. While we, as the authors and developers of the IFM framework, may have unconsciously looked more favorably onto the IFM framework, we tried to base our evaluation, wherever possible, on the concrete models created. Furthermore, we provided references to complementing studies on

Table 10. Comparison of effects of premature commitments

Characteristic	Function Structures	IFM Framework
Premature commitment	The function structure itself does not support indication of assumptions. Even though it does not force designers to make premature decisions, inexperienced users might be tempted to do so (unconsciously).	The model can be gradually elaborated as knowledge increases. Because annotation is possible, assumptions can be indicated as such.

the use of the IFM framework where appropriate in order to support our assessments. This includes applications to more complex systems and other types of modeling challenges. We hope to have been able to provide a compelling and critical comparison of both modeling approaches that may trigger interest in the wider research community.

Benchmarking of function modeling approaches is a new endeavor. The currently used benchmarking protocol will need refinement after it has been applied to several modeling approaches. A challenge in using the protocol has been the partial overlap of some characteristics, which belong to different categories, and their explanations, which sometimes leave room for interpretation. We complemented the proposed set of reasoning characteristics by a set of modeling perspectives in order to enable a more precise comparison of the modeled content. During the analysis, we identified further criteria that would be relevant for a benchmark, such as the effort required for learning a particular modeling approach, effort (time) required for creating models, and maintenance of fluency in solution finding (avoiding early fixation). We expect that the use of the benchmarking protocol in other comparisons will provide additional feedback, maybe even contradicting our observations.

7. CONCLUSION

It is possible that no other type of design method has produced such a variety of modeling approaches as function modeling. Each of these approaches has specific merits for modeling or analyzing products, services, or their combination (Wallace, 2011; Daalhuizen, 2014). Yet their specific advantages are rarely made explicit. This begs the question of how designers should choose between approaches when facing a specific design task. Modeling approaches differ in terms of the prevalent design context targeted, novelty or complexity of a design task, as well as stages of a design project in which they are particularly useful (e.g., product planning as opposed to conceptual or embodiment design).

In this article, we present a comparative study of two function modeling approaches; on the one hand, as the probably most widespread approach in engineering design literature, function structures after Pahl et al. (2007), and on the other hand, the more recent IFM framework proposed by Eisenbart et al. (2016a). Both are exemplarily applied to model the example of a glue gun, which has been similarly used in preceding work seeding this Special Issue. The approaches are compared using the benchmarking protocol by Summers et al. (2013) to enhance comparability of our results with other scholars' in the context of this Special Issue. Particularly, we comparatively investigated the approaches pertaining to representation, modeling, cognitive dimension, and reasoning characteristics following the protocol. Further points of comparison have been included to add depth to the presented analysis, going beyond the example of the glue gun.

Based on the conducted benchmarking, concrete strengths and weaknesses in applying the approaches were derived.

Many of these are expected to similarly apply beyond the given example. It is obvious that the models differ in terms of their comprehensiveness, which directly translates into the modeling efforts and prerequisites for applying them. Function structures have clear advantages over the IFM framework here, as they are less comprehensive in terms of addressed contents and thus can be modeled more quickly. Yet, in turn, the generated IF model incorporates information saliently that is rather intangible in function structures. This particularly pertains to linking it with the (physical) structure of the modeled entity. By extension, the assumptions and implicit decisions made by designers during the design process, which gradually constrain the solution space, can become more visible. These might be crucial to enable others who were not involved in the model generation process to comprehend the model at a later stage. Moreover, the inherent consistency check enforced by the IFM framework, which is almost entirely missing in function structures, triggers a more conscious modeling process and level of abstraction used within the model. This is not specifically supported in function structures and the main, subfunctions, and auxiliary functions can be modeled within the same model indiscriminately. This is something Eckert et al. (2010) and Alink (2010) similarly found and discuss to pose difficulties in the practical application of function structures. Eventually, it has to be noted that while the IFM framework provides advantages in the support of function modeling and analysis for complex, interdisciplinary systems, modeling efforts may increase fairly quickly given that this involves diverse types of information to be modeled. The complexity of the system under development has an impact on the perceived benefit of function modeling. With reference to the example of the glue gun, the level of detail and sheer amount of information incorporated in the IF model may not be necessary for most design purposes. Yet abstract models such as function models unfold their potential, that is, create benefit, primarily when used for modeling complex or highly complicated systems. Therefore, we believe that while function structures provide benefits in terms of simplicity, the IFM framework has the potential to excel over them in terms of support for consistency, comprehensiveness, and analysis options. With this in mind, one may expect function structures to be advantageous whenever information is limited, such as in new design projects, while the IFM framework is particularly valuable for redesign projects that can use existing knowledge about a system as a starting point.

We hope that the comparison presented here will spark interest in the design research community and industrial practice regarding the practical application and respective benefits of the approaches in different application scenarios. From such a discussion, we hope to be able to distill further insight and potential improvement of the IFM framework and function structures. We see vast opportunities for further cross-fertilization in this endeavor with other function modeling approaches discussed in this Special Issue and beyond.

REFERENCES

- Alink, T. (2010). *Bedeutung, Darstellung und Formulierung von Funktionen für das Lösen von Gestaltungsproblemen mit dem C&C-Ansatz*. PhD Thesis. Institut für Produktentwicklung, Karlsruhe Institute of Technology.
- Andreasen, M.M., Hansen, C.T., & Cash, P. (2015). *Conceptual Design: Interpretations, Mindsets and Models*. Heidelberg: Springer.
- Aurisicchio, M., Bracewell, R., & Armstrong, G. (2012). The function analysis diagram. *Proc. ASME Design Engineering Technical Conf. Computers and Information in Engineering Conf.*, Chicago, August 12–15.
- Aurisicchio, M., Eng, N., Ortiz, N.J., Childs, P., & Bracewell, R. (2011). On the functions of products. *Proc. 18th Int. Conf. Engineering Design*. Glasgow: Design Society.
- Blessing, L.T.M., & Upton, N. (1997). A methodology for preliminary design of mechanical aircraft systems. *Proc. AIAA/SAE World Aviation Congr.* Reston, VA: American Institute of Aeronautics and Astronautics.
- Booth, J.W., Reid, T.N., Eckert, C., & Ramani, K. (2015). Comparing functional analysis methods for product dissection tasks. *Journal of Mechanical Design* 137, 081101-1–081101-10.
- Brown, D.C., & Blessing, L.T.M. (2005). The relationship between function and affordance. *Proc. ASME Int. Design Engineering Technical Conf. Computers and Information Engineering Conf.*, Long Beach, CA, September 24–28.
- Caldwell, B.W., Sen, C., Mocko, G.M., & Summers, J.D. (2011). An empirical study of the expressiveness of the functional basis. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 25(3), 273–287.
- Carrara, M., Garbacz, P., & Vermaas, P. (2011). If engineering function is a family resemblance concept: assessing three formalization strategies. *Applied Ontology* 6(2), 141–163.
- Chakrabarti, A., & Bligh, T.P. (2001). A scheme for functional reasoning in conceptual design. *Design Studies* 22(6), 493–517.
- Chandrasekaran, B. (2005). Representing function: relating functional representation and functional modelling research streams. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 19(2), 65–74.
- Chandrasekaran, B., & Josephson, J. (2000). Function in device representation. *Engineering With Computers* 16(3–4), 162–177.
- 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.
- Chittaro, L., & Kumar, A.N. (1998). Reasoning about function and its applications to engineering. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 12(4), 331–336.
- Crilly, N. (2010). The role that artefacts play: technical, social and aesthetic functions. *Design Studies* 31(4), 311–344.
- Crilly, N. (2015). The proliferation of function: multiple systems playing multiple roles in multiple supersystems. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 29(1), 83–92.
- Cross, N. (2008). *Engineering Design Methods: Strategies for Product Design*. Chichester: Wiley.
- Daalhuizen, J. (2014). *Method usage in design: how methods function as mental tools for designers*. PhD Thesis. Delft University of Technology.
- Deng, Y. (2002). Function and behaviour representation in conceptual mechanical engineering. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 16(5), 343–362.
- Dewey, A. (2000). Digital and analogue electronic design automation. In *The Electrical Engineering Handbook* (Dorf, R.C., Ed.). Boca Raton, FL: CRC Press.
- Dohr, F., Eisenbart, B., Huwig, C., Blessing, L.T.M., & Vielhaber, M. (2014). Software support for the consistent transition from requirements to functional modeling to system simulation. *Proc. 10th NordDesign Conf.*, Espoo, Finland, August 27–29.
- 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., & Albers, A. (2010). Issue driven analysis of an existing product at different levels of abstraction. *Proc. 11th Int. Design Conf.* Glasgow: Design Society.
- Eder, W., & Hosnedl, S. (2008). *Design Engineering: A Manual for Enhanced Creativity*. Boca Raton, FL: CRC Press.
- Eisenbart, B., Blessing, L.T.M., & Gericke, K. (2012). Functional modelling perspectives across disciplines: a literature review. *Proc. 12th Int. Design Conf.* Glasgow: Design Society.
- Eisenbart, B., Dohr, F., Gericke, K., Vielhaber, M., & Blessing, L.T.M. (2013). Potentials for realising a consistent transition between function modelling with the IFM framework and early system simulation. *Proc. 19th Int. Conf. Engineering Design*, Seoul, South Korea, August 19–22.
- Eisenbart, B., Gericke, K., & Blessing, L.T.M. (2013a). An analysis of functional model approaches across disciplines. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 27(3), 281–289.
- Eisenbart, B., Gericke, K., & Blessing, L.T.M. (2013b). Adapting the IFM framework to functional approaches across disciplines. *Proc. 19th Int. Conf. Engineering Design*, Seoul, South Korea, August 19–22.
- Eisenbart, B., Gericke, K., & Blessing, L.T.M. (2015). DSM for modeling and analyzing functionality: views of practitioners. *Proc. 17th Int. DSM Conf.*, Fort Worth, TX, November 4–6.
- Eisenbart, B., Gericke, K., & Blessing, L.T.M. (2016a). A DSM-based framework for integrated function modeling: concept, application and evaluation. *Research in Engineering Design* 28(1), 25–51.
- Eisenbart, B., Gericke, K., & Blessing, L.T.M. (2016b). Taking a look at the utilisation of function models in interdisciplinary design: insights from ten engineering companies. *Research in Engineering Design* 28(3), 299–331.
- Eisenbart, B., Mandel, C., Gericke, K., & Blessing, L.T.M. (2015). Integrated function modelling: comparing the IFM framework with SysML. *Proc. 20th Int. Conf. Engineering Design*. Glasgow: Design Society.
- Eppinger, S.D., & Browning, T.R. (2012). *Design Structure Matrix Methods and Applications*. Cambridge, MA: MIT Press.
- Erden, M., Komoto, H., van Beek, T.J., D'Amelio, V., Echavarría, E., & Tomiyama, T. (2008). A review of function modeling: approaches and applications. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 22(2), 147–169.
- Ericson, A., & Larsson, T. (2005). A service perspective on product development. Towards functional products. *Proc. 12th Int. Product Development Management Conf.*, Copenhagen.
- Far, B.H., & Elamy, H. (2005). Functional reasoning theories: problems and perspectives. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 19(2), 75–88.
- Garbacz, P., Borgo, S., Carrara, M., & Vermaas, P.E. (2011). Two ontology-driven formalisations of function and their comparison. *Journal of Engineering Design* 22(11–12), 733–764.
- Gausemeier, J., Frank, U., Donoth, J., & Kahl, S. (2009). Specification technique for the description of self-optimizing mechatronic systems. *Research in Engineering Design* 20(4), 201–223.
- Gericke, K., Meissner, M., & Paetzold, K. (2013). Design context. *Proc. 19th Int. Conf. Engineering Design*. Glasgow: Design Society.
- Gericke, K., Qureshi, A.J., & Blessing, L.T.M. (2013). Analyzing transdisciplinary design process in industry. An overview. *Proc. ASME Int. Design Engineering Technical Conf. Computer and Information in Engineering Conf.*, Portland, OR, August 4–7.
- Goel, A. (2013). One thirty year long case study: fifteen principles implications of the AI methodology for functional modelling. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 27(3), 203–215.
- Houkes, W., & Vermaas, P.E. (2010). Technical functions: on the use and design of artefacts. In *Philosophy of Engineering and Technology*. Dordrecht: Springer-Verlag.
- Kreimeyer, M., & Lindemann, U. (2011). *Complexity Metrics in Engineering Design: Managing the Structure of Design Processes*. Berlin: Springer.
- Kruchten, P. (2004). *The Rational Unified Process: An Introduction*. Upper Saddle River, NJ: Pearson Education.
- Kurfman, M., Stock, M., & 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), 682–639.
- Kurfman, A., Stone, R.B., Rajan, J.R., & Wood, K.L. (2001). Functional modeling experimental studies. *Proc. ASME Design Engineering Technical Conf.*, Pittsburgh, PA, September 9–12.
- Lindemann, U., Maurer, M., & Braun, T. (2009). *Structural Complexity Management*. Berlin: Springer.
- Maher, M., & Tang, H. (2003). Co-evolution as a computational and cognitive model of design. *Research in Engineering Design* 14(1), 47–63.
- Maier, J.R., & Fadel, G.M. (2001). Affordance: the fundamental concept in engineering design. *Proc. ASME Int. Design Engineering Technical Conf. Computer and Information in Engineering Conf.*, Pittsburgh, PA, September 9–12.
- Müller, P., Schmidt-Kretschmer, M., & Blessing, L.T.M. (2007). Function allocation in product-service-systems: are there analogies between PSS and mechatronics? *Proc. AEDS Workshop.*, pp. 47–556, Pilsen, Czech Republic.

- Pahl, G., Beitz, W., Feldhusen, J., & Grote, K.H. (2007). *Engineering Design: A Systematic Approach*. Berlin: Springer-Verlag.
- Roozenburg, N.F.M., & Eekels, J. (1995). *Product Design: Fundamentals and Methods*. Chichester: Wiley.
- Scheffer, L., Lavagno, L., & Martin, G. (2006). *EDA for Implementation, Circuit Design, and Process Technology*. Boca Raton, FL: CRC Press.
- Sen, C., Summers, J.D., & Mocko, G.M. (2013). A formal representation of function structure graphs for physics-based reasoning. *Journal of Computational and Information Science in Engineering* 13(2).
- Shostack, G.L. (1982). How to design a service. *European Journal of Marketing* 16(1), 49–63.
- Srinivasan, V., Chakrabarti, A., & Lindemann, U. (2012). A framework for describing functions in design. *Proc. 12th Int. Design Conf.* Glasgow: Design Society.
- Stone, R.B., & Wood, K.L. (2000). Development of a functional basis for design. *Journal of Mechanical Design* 122(4), 359–370.
- Summers, J.D., Eckert, C., & Goel, A.K. (2013). Function in engineering: benchmarking representations and models. *Proc. Int. Conf. Engineering Design*. Glasgow: Design Society.
- Tomiyama, T., Gu, P., Jin, Y., Lutters, D., Kind, C., & Kimura, F. (2009). Design methodologies: industrial and educational application. *CIRP Annals—Manufacturing Technology* 58(2), 543–565.
- Tomiyama, T., van Beek, T.J., Alvarez Cabrera, A.A., Komoto, H., & D'Amelio, V. (2013). Making function modeling practically usable. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 27(8), 301–309.
- Tukker, A., van den Berg, C., & Tischner, U. (2006). Product-services: a specific value proposition. In *New Business for Old Europe* (Tukker, A., & Tischner, U., Eds.). Sheffield: Greanleaf.
- Ullman, D. (2010). *The Mechanical Design Process*. Boston: McGraw-Hill Higher Education.
- Ulrich, K., & Eppinger, S.D. (2008). *Product Design and Development*. New York: McGraw-Hill Higher Education.
- US Department of Defense. (2001). *Systems Engineering Fundamentals*. Fort Belvoir, VA: Defense Acquisition University Press.
- van Beek, T.J., & Tomiyama, T. (2009). Connecting views in mechatronic systems design, a functions modelling approach. *Proc. ASME 2009 Int. Conf. Mechatronic and Embedded Systems and Applications*, Beijing, October 12–15.
- VDI. (2004). *VDI 2206—Design Methodology for Mechatronic Systems*. Berlin: Beuth Verlag.
- Vermaas, P. (2009). The flexible meaning of function in engineering. *Proc. 17th Int. Conf. Engineering Design*. Glasgow: Design Society.
- Vermaas, P. (2011). Accepting ambiguity of engineering functional descriptions. *Proc. 18th Int. Conf. Engineering Design*. Glasgow: Design Society.
- Vermaas, P. (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), 191–202.
- Wallace, K. (2011). Transferring design methods into practice. In *The Future of Design Methodology* (Birkofer, H., Ed.), pp. 239–248. London: Springer.
- Warell, A. (1999). Introducing a use perspective in product design theory and methodology. *Proc. ASME Int. Design Engineering Technical Conf. Computer and Information in Engineering Conf.* New York: American Society of Mechanical Engineers.
- Watanabe, K., Mikoshiba, S., Tateyama, T., Shimomura, Y., & Kimita, K. (2011). Service design methodology for cooperative services. *Proc. ASME Int. Design Engineering Technical Conf. Computer and Information in Engineering Conf.*, Washington, DC, August 28–31.
- Weber, C. (2007). Looking at “DFX” and “product maturity” from the perspective of a new approach to modelling product and product development processes. *Proc. 17th CIRP Design Conf.*, pp. 85–104. Berlin: Springer.
- Weilkiens, T. (2008). *Systems Engineering With SysML: Modelling, Analysis, Design*. Heidelberg: Springer-Verlag.

Kilian Gericke is a Research Scientist at the University of Luxembourg. He attained his PhD in design management at Technische Universität, Berlin. Dr. Gericke's research interests are in design process modeling and management, design methodology, and function modeling.

Boris Eisenbart is a Senior Lecturer in product design engineering at Swinburne University of Technology in Melbourne. He holds a Diploma in mechatronics engineering from Saarland University and a PhD in engineering design from the University of Luxembourg. Before his current role, Dr. Eisenbart was an Assistant Professor in product innovation management at TU Delft and worked as a Postdoctoral Research Associate in design thinking and strategy at the University of Sydney.