INTERNATIONAL DESIGN CONFERENCE - DESIGN 2012
Dubrovnik - Croatia, May 21 - 24, 2012.

A FRAMEWORK FOR DESCRIBING FUNCTIONS IN DESIGN

V. Srinivasan, A. Chakrabarti and U. Lindemann

Keywords: function, framework, system, environment, SAPPhIRE, requirement, solution, intended, unintended

1. Introduction and objective

In designs and designing, function is an important entity. During designing, designers develop functions, and develop designs at several levels of abstraction to satisfy the functions. If designs do not satisfy functions, the designs or/and the functions are modified for conformance between them. Knowledge of functions is important for modelling (geometric, kinematic, dynamic, etc.), generation, modification, exploration, visualization, explanation, evaluation, diagnosis, and repair of designs and processes [Chakrabarti and Blessing 1996], [Deng et al. 2000], [Stone and Chakrabarti 2005].
In spite of function being a subject of research for many years and being widely used in various domains and research areas, several issues exist with its understanding and usage. Researchers attribute different meanings to function and sometimes use it interchangeably with behaviour [Chittaro and Kumar 1998], purpose and operation [Ullman 1992]. The definitions of function are both, specific and generic, and there is no clear and uniform definition because function depends on designer's intuition [Umeda et al. 1996]. This subjective nature is also acknowledged by Balachandran and Gero [Balachandran and Gero 1990], Keuneke [Keuneke 1991], Chandrasekaran and Josephson [Chandrasekaran and Josephson 2000], and Deng et al. [Deng et al. 2000]. Function lacks a single precise meaning but has a number of co-existing meanings and all of these are used,but the different descriptions are ambiguous, create confusion in communication and archiving, and obstruct teaching and formalization [Vermaas 2011]. On the other hand, having flexibility in the meaning of function has its advantages in engineering [Vermaas 2009]. Chakrabarti and Blessing [Chakrabarti and Blessing 1996] attribute the difficulties to the multiple ways of interpretation of function – abstraction of intended behaviour of a design, indexing of intended behaviour, relationship between a design and its environment, external behaviour of a design or its internal behaviour, etc. Attempts to identify a single meaning for function which has many meanings, bound to fail [Chandrasekaran and Josephson 2000]. Many definitions of function do not have clear relationships between them [Hubka and Eder 2001], [Stone and Chakrabarti 2005], [Kitamura and Mizoguchi 2010].
To summarise, researchers from various fields of design research propose definitions of function as viewed by them in their fields. Therefore, the different definitions contain different views of function. Not all these views are made explicit in all the definitions. This implicitness of the different views reduces common understanding among researchers in different fields. Therefore, providing a common set of views to describe the different definitions of function can provide a better shared understanding.
The overall aim of this research is to develop and validate a framework that contains explicitly the different views of function in design. The framework enables describing the different definitions of function in terms of the views. We believe that doing this provides a common basis for a better comparison between the definitions to assess the similarities and differences between the functions. Further, a better comparison will also enable greater common understanding and usage of functions.

2. Research methodology

The following methodology is used for this research:

a) Review literature: Various definitions and representations of function from literature are reviewed, and are supplemented using examples from the literature for a better understanding.
b) Identify views: The different views in functions are identified from the literature review. The views are deduced from the definitions, representations and examples.
c) Propose framework: A framework is proposed based on the identified views and earlier work.
d) Validate framework: The framework is validated theoretically by showing that it can be used to describe the functions using the views.

3. Literature review and identification of views

In this section, the various definitions, representations and examples of function from literature are reviewed. We assume that all the definitions, representations and examples are correct.

Rodenacker [Rodenacker 1971] defines function as the transformation from input to output; input and output can be types of material, energy and signal. For example, the function of a coffee mill is to convert coffee beans, electrical energy and electrical signals into coffee powder, heat energy and electrical signals.

Miles [Miles 1972] expresses function as *to do something*. The function is represented using a verb and a noun. For example, provide light, pump water, indicate time, etc. Miles makes a distinction between primary and secondary functions. For example, the primary function of a domestic pump is to pump water and its secondary function is to operate at low noise-level.

In the Function-Behaviour-Structure model of designing, Gero [Gero 1990] considers functions to fulfil the expectations of the purposes of the resulting artefact. For example, when designing windows, some functions are provision of daylight, control of ventilation, etc. In his design prototype schema, he represents functions using combination of verbs, nouns and adjectives (e.g., provide_daylighting, ventilation_control, etc.). In his model, he identifies two kinds of behaviours – intended and actual, which are derived from function and structure, respectively. However, the actual behaviour is not translated into function.

Kenuneke [Kenuneke 1991] defines function as the intended purpose of a device. The function describes the device's goals at a level of abstraction that is of interest at the device level. Functions are achieved by behaviours. Functions are specified by the goals of activities of devices and components, and behaviors are represented as a causal sequence of states. Keuneke identifies the following different types of expected functions: *ToMake*, *ToMaintain*, *ToPrevent* and *ToControl*. These functions help to achieve a specific partial state, achieve and sustain a desired state, keep a system out of an undesired state and, regulate state changes, respectively.

Ullman [Ullman 1992] defines function as a logical flow of energy, material, or information between objects or, the change of state of an object caused by one or more of the flows. Ullman represents the overall function in terms of a box with inputs and outputs of energy, material and information. For example, the function of a one-handed bar clamp is to transform the grip force of one hand to a controllable force capable of clamping objects together. The function is represented using action verbs. Overall function is decomposed into sub-functions.

Pahl and Beitz [Pahl and Beitz 1996] define function as the intended input-output relationship of a system whose purpose is to perform a task; the task is the conversion of input to output; and function is an abstraction of the task. The input and the output can be material, energy and signal. Pahl and Beitz divide the solution-neutral overall function into sub-functions, and these together satisfy the overall function. The sub-functions are connected to each other in a way that the output from one sub-function becomes the input to the next, and so forth, and a temporal ordering is maintained. Function is represented using a verb and a noun. For example, in a fuel gauge the overall function is to measure and indicate liquid quantities; this overall function is divided into sub-functions: receive signal, change signal, correct signal, channel signal and indicate signal.

In the Function-Behaviour-State (FBS) model, Umeda et al. [Umeda et al. 1996] define function as a description of behaviour abstracted by human through recognition of behaviour in order to utilize it. Behaviours are defined as sequential state transitions with time, and are specified using physical phenomena. Umeda et al. feel that it is difficult to represent function independent of behaviour from which it is abstracted, so function is represented as a combination of "to do something" and a set of behaviours that exhibit this function. For instance, the behaviours of ringing a bell or oscillating a string exhibit the function of producing sound. Functions are causal or task decomposed into sub-functions. For example, to generate light is causally and task decomposed into: light a lamp with electricity and generate electricity. Function is represented using verb, object and modifier (adjective). The FBS-modeller helps detect unintended physical phenomena but these are not abstracted into unintended functions.

Simon [Simon 1996] proposes that the fulfilment of a goal or adaptation to a purpose of an artefact involves the relationship between the purpose or goal of the artefact, the character of the artefact and the environment in which the artefact performs. For example, the purpose of a clock is to tell time, can be described in terms of the gears, pendulum, forces of spring and gravity on pendulum, and needs a sunny environment (e.g., sundials). Simon considers an artefact as an interface between an inner environment (organization and substance of artefact) and outer environment (surroundings in which artefact performs). According to Simon, an artefact will achieve its purpose only if the inner environment matches the outer environment. For example, a theory of the airplane draws on natural science for an explanation of its inner environment (for example, power plant), its outer environment (character of atmosphere at different altitudes), and the relation between its inner and outer environments (movement of an aerofoil in a medium).

Chakrabarti [Chakrabarti 1998] proposes two views of function. In one view, function is taken as intended behaviour and at the same level of abstraction as behaviour. This view of function and behaviour are described using state variables and components within the system boundary. The intended behaviour could be a subset or an aggregate of a subset of expected behaviour. For example, an intended behaviour of a door latch is to press a door handle to cause a wedge to retract. This may be achieved by the expected behaviour of a chain of components which together transform the motion of the handle into that of the wedge. In the other view, function is taken at a higher level of abstraction and is referred to as purpose. For example, locking a door against wind. In this work, Chakrabarti distinguishes function from behaviour by treating function as intentional and behaviour as actual or expected.

Chittaro and Kumar [Chittaro and Kumar 1998] define function as the roles played by components in a system. In other words functions convey what components do. For example, a switch in an electrical lamp is seen as a barrier or conduit for electrical current. Chittaro and Kumar add the dimension of intentions of designers or users through teleology, which is defined as the purposes assigned by designers or users to a system and the operational conditions required to meet them. In other words, this knowledge answers the question why is a component needed in the system. For example, a switch in an electrical lamp may be used to turn on or off the lamp.

Chandrasekaran and Josephson [Chandrasekaran and Josephson 2000] make a distinction between the environment- and device-centric views of function. In the former, function is viewed as an intended effect that a device has on its environment, and is also referred to as function as effect. For example, the function of a buzzer is to provide a means by which a person at one location may cause sound to be produced at another location. In the latter, function is defined in terms of the attributes and properties of the components that it is made of. For example, the function of a buzzer is to make sound come from a box when a switch in another box is closed. Chandrasekaran and Josephson also introduce the terms, mode of deployment and role. Mode of deployment is used to capture the notion of how a device is used so as to produce the intended effect. It describes the connections between a device and its environment, and the actions taken by the device on its environment. For example, a battery's function is to provide voltage between two terminals under the mode of deployment that there are electrical connections to the battery's terminals. Role is a term used for capturing an aspect of effect that a device has on its environment. For example, the heart plays the role of a pump, the role of moon is to cause tides, etc.

Wood and Greer [Wood and Greer 2001] define function as what a product must do. For the purpose of modelling and representation, function is taken as a general input-output relationship i.e., an action of the product on its inputs to produce outputs [Stone and Wood 2000]. Input and output can be material, energy and signal. Stone and Wood decompose the product function into sub-functions to represent more elementary tasks of the product. Both, product- and sub-function are represented using a verb and an object. For example, convert torque, transmit electricity, etc. Further, they define function as a description of an operation to be performed by the product, and express it as the verb of a sub-function. Flow is defined as a change in material, energy and signal with time, and is expressed as the object of a sub-function. The function and flow are illustrated through Functional Basis, a design ontology for sub-functions.

Hubka and Eder [Hubka and Eder 2001, 2002] define function as the required or desired, internal and cross-boundary, capabilities of a future/existing real system that will make it possible for the system to perform its intended goal tasks. They describe function as the transformation of the desired and secondary inputs into the desired and secondary outputs, within the real system. Functions are formulated using a combination of verbs and nouns, words or phrases. They categorize functions based on degree of complexity, degree of abstraction and purpose.

Deng [Deng 2002] classifies function into purpose function and action function. According to Deng, these functions are at different levels of abstraction, and therefore, can exist simultaneously in the same design. Action function is an abstraction of intended behaviour. Action is defined as an interaction between two objects which may be components of a system, or the system and its environment. Purpose function is a description of a designer's intention or the purpose of a design. It is thus abstract and subjective. Deng considers the overall function and some of its sub-functions at the upper level of designing as purpose functions. The lower-level functions that implement the purpose function are referred to as action functions. For example, in a time-telling device, "to tell time" is the purpose function, which can be embodied by the action function, "to rotate the hour, minute and second hands about a pivot at a specific angular velocity."

In the SBF model of Goel et al. [Goel et al. 2009], function is specified as the transition of an intended input to an intended output with a reference to behaviour(s) that satisfy the function. A schema that specifies the function also includes conditions under which a behaviour achieves the function. For example, a gyroscope takes as input an angular momentum of particular magnitude in clockwise direction at the input location, and delivers a proportional angular momentum as output in clockwise direction at the output shaft location. This transformation is achieved through the behaviour, "TransferAngularMomentum". Behaviour in this model is represented as a sequence of states and transitions between them. The states specify the evolution in values of substances and/or components. Continuous state variables are discretised and temporal order is taken into account in causal ordering. Each state transition is annotated by physical laws, mathematical equations, functions of sub-systems, structural constraints, other behaviours, or state/transition.

Kitamura and Mizoguchi [Kitamura and Mizoguchi 2010] make a distinction between actual function and capacity function. Actual function is a function that is directly related to a physical process performed by an artefact. Capacity function is a feature or characteristic of an artefact that exists inside the device, and expresses the capacity to perform an actual function. Kitamura and Mizoguchi also make a distinction between artefact function and device function. The former is an actual function intended by a user when an artefact is used externally by a user. The word external here is used to mean external to the device. This function depends on the user's intentions. For example, the function of a screwdriver is to perform a screwing function or a hitting function. The device function refers to a function performed by a component in a system that contributes to the overall function of the system. This function depends on the functional hierarchy of the system. For example, the purpose of a heat exchanger in a power plant whose purpose is to convert heat energy into electrical energy, is to give heat. A distinction is also made between essential external function and accidental external function. An essential function of a system is one that is intended by the designers of the system. For example the screwing function of a screwdriver. An accidental function of a system is a function that is not intended by the designers of the system. For example, the hitting action of the screwdriver. According to Kitamura and Mizoguchi, the accidental functions are affordances due to the capacity functions of the system. The set of properties that help perform the essential external function also afford the system to perform the unintended, external function.

The following views of function are identified from the literature:

a) Level of abstraction view: The literature shows that functions are addressed at different levels of abstraction. For example, function of [Miles 1972], [Gero 1990], [Keuneke 1991], [Pahl and Beitz 1996], [Wood and Greer 2001], etc. are defined as, "to do something" or the purpose of an artefact. On the other hand, function of [Rodenacker 1971], [Stone and Wood 2000], [Hubka and Eder 2002], etc. are defined as the transformation of inputs to outputs. We consider this function as less abstract than the previous one because to do something or the purpose is an interpretation of the transformation. [Chakrabarti 1998] and [Deng 2002] explicitly state the difference in their functions in terms of the level of abstraction. The actual function of [Kitamura and Mizoguchi 2010] is less abstract than their purpose function because the purpose function uses actual function (properties and characteristics of artefact) to achieve the artefact's purpose. The difference in levels of abstraction among functions is also made clear in terms of the information used to describe the functions. For example, the overall function of Pahl and Beitz is more abstract than their sub-functions because the former is more solution-neutral than the latter; the environment-centric function of [Chandrasekaran and Josephson 2000] is more abstract than their device-centric function because the latter contains more information about the device (system) than the former.
b) Requirement-Solution view: Majority of functions are considered as requirements which need to be fulfilled. For example, function of [Gero 1990], (overall) [Pahl and Beitz 1996], (overall) [Stone and Wood 2000], (purpose) [Deng 2002], [Goel et al. 2009], etc. Some functions are also considered as solutions. For example sub-functions in [Ullman 1992], [Pahl and Beitz 1996], [Umeda et al. 1996], [Stone and Wood 2000], etc. are treated as solutions for the overall function because they together satisfy the overall function. The action function of [Deng 2002] is also seen as a solution for the purpose function.
c) System-Environment view: Although literature describes functions as pertaining only to system (i.e., device, product, artefact, etc.), functions are described as causing effects on the environment (i.e. surroundings of device, product, etc.) and use information of both system and its environment at several levels of abstraction. Some definitions of function focus on the system whereas others focus on its environment. For example, the functions of [Gero 1990], (environment-centric) [Chandrasekaran and Josephson 2000], (higher level) [Chakrabarti 1998], etc. describe the effects of system on environment. On the other hand, the functions (secondary) of [Miles 1972], (device-centric) of [Chandrasekaran and Josephson 2000], (lower-level) [Chakrabarti 1998], etc. focus on the system, and also contain more information about the system. In functions, relationships between system and environment are also described. For example, [Rodenacker 1971], [Hubka and Eder 2002], etc., describe the relationships in terms of the inputs and outputs that enter and leave the system boundary as material, energy and information; mode of deployment of [Chandrasekaran and Josephson 2000] is defined as relationships between system and environment in terms of connections of components and parts. In functions, system hierarchy with relationships among them are also explained. For example [Ullman 1992], [Umeda et al. 1996], [Pahl and Beitz 1996], etc. divide the (overall-) function into sub-functions; these sub-functions have relationships of type temporal, causal, etc. Another example for system hierarchy is the artefact and the device functions of [Kitamura and Mizoguchi 2010] as these functions pertain to the system and the components that constitute the system, respectively.
d) Intended-Unintended view: Functions in the literature are intended or unintended by designers or users. For example, the functions of [Rodenacker 1971], (primary) [Miles 1972], [Gero 1990], [Pahl and Beitz 1996], (higher level) [Chakrabarti 1998], [Goel et al. 2009], (essential) [Kitamura and Mizoguchi 2010], etc. are intended by designers. On the other hand, functions of (secondary) [Miles 1972], (secondary) transformations of [Hubka and Eder 2002], (accidental) [Kitamura and Mizoguchi 2010], etc. are unintended by designers. The functions of (lower level) [Chakrabarti 1998] and (capacity) [Deng 2002] can be intended or unintended by designers or users.

4. Proposal of framework

The views identified in the previous section are related to earlier research and are explained as follows. The SAPPhIRE (**S**tate change, **A**ction, **P**art, **Ph**enomenon, **I**nput, o**R**gan, **E**ffect) model of causality is developed to explain the working of biological and engineered systems [Chakrabarti et al. 2005]. Srinivasan and Chakrabarti [Srinivasan and Chakrabarti 2009] define the constructs as follows. Phenomenon is an interaction between a system and its environment. For example, movement of a body, transfer of heat from a body to its surroundings, etc. Effect is a principle underlying the interaction. For example, the second equation of motion, $x = u \times t + 0.5 \times a \times t^2$, where x is the displacement, u is the initial velocity, a is the acceleration and t is the time duration; the rate of heat transfer, $Q = h \times A \times \Delta T$, where h is the convective heat transfer co-efficient, A is the surface area of the body, ΔT is the temperature difference between body and surroundings, etc. Input is a physical quantity in the form of material, energy or signal that is responsible for the interaction. For example, acceleration on a body, temperature difference between a body and its surroundings, etc. Organ is a set of properties and conditions of the system and its environment that are also responsible for the interaction. For example, body has degrees of freedom to move in certain directions, body is initially at rest (u=0 at t=0), etc. Part is a set of components and interfaces that together make the system and its environment. For example, body lying on a floor, body held in an air medium, etc. State change is a change in property of the system due to the interaction. For example, change in spatial position of the body, change in heat energy stored in body, etc. Action is a higher level abstraction or interpretation of the interaction. For example, transport body, cool body, etc. The SAPPhIRE model is empirically found to be able to describe both, analysis and synthesis of engineered systems [Srinivasan and Chakrabarti 2009]. The model is also empirically validated and found to be able to describe outcomes in designing [Srinivasan and Chakrabarti 2010].

Requirement is defined as an expression of what a design should have; solution is defined as a means for fulfilling requirements [Srinivasan and Chakrabarti 2010]. A co-evolving requirement-solution model is proposed as a part of the integrated model of designing, GEMS of SAPPhIRE as req-sol [Srinivasan and Chakrabarti 2010]; the GEMS (Generate-Evaluate-Modify-Select) activity model and the SAPPhIRE outcome model are the other parts of the integrated model. An empirical validation of the req-sol model shows that requirements and solutions exist at several levels of abstraction and both these co-evolve during designing [Srinivasan and Chakrabarti 2010]. This view enables the distinction of outcomes as requirements or solutions.

Based on an analysis of existing design theories, models and approaches, Ranjan et al. [Ranjan et al. 2012] propose the system-environment model of designing. This model comprises the constructs: element, sub-system, system, environment and relationships between them. These constructs are defined as follows. A system is an overall entity at any level of abstraction. A sub-system is a subset of a system. An element is a basic entity of a system or a sub-system which cannot be further sub-divided. An environment refers to all subsets of the universe apart from the system within which the system must work. The relationships between these constructs can be explained as follows. Elements combine together to comprise sub-systems. Sub-systems and elements combine together to comprise a system. Empirical validation of this model using protocol studies of design sessions where this model was not explicitly followed, reveals that the constructs of the system-environment model are present in designing at different levels of abstraction. The validation also reveals a co-evolution between system and environment.

Brown and Blessing [Brown and Blessing 2005] discuss the relationship between function and affordance. Maier and Fadel [Maier and Fadel 2003] define affordances as the set of interactions between the artefact and user in which the properties of the artefact are or maybe perceived by the user as potential uses. For example, a pen causes a hole in a paper when pressed against the paper with enough force. When a device satisfies some behavioural constraints in a particular mode of deployment in an environment, then the device plays a role in that environment. Some roles are intended by the designer and some are intended by the user; the remaining roles are unintended. If a role is intended by an agent (user or designer), then that role satisfies a behavioural constraint, and this behavioural constraint is said to provide a function for the device in that environment. For example,

the pen writes on the paper. In cases, where the role intended by the designer matches with the role intended by the user, then the device is said to perform the intended function. In other cases where the role is intended but was not intended by the designer, the device still provides a function intended by the user. For example, a pen for punching holes.
The literature review in this section reveals a relationship with the views of descriptions of functions identified in the earlier section. The literature in this section comprising the SAPPhIRE model, the requirement-solution model, system-environment model and intended-unintended views, is used as a basis for the framework.

5. Theoretical validation of framework

In this section, the framework is validated by using it to describe the functions reviewed earlier. Rodenacker considers conversions as requirements which can be intended (coffee beans to coffee powder) or unintended (input energy to heat energy) by designers. His function pertains to the system and shows relationship with environment through input and output. The input and output are similar to input of the SAPPhIRE model in a sense that both these transit the system boundary, and can be expressed in terms of material, energy and signal. But the conversion from input to output (coffee beans to coffee powder) resembles state change (non-powder to powder) of the SAPPhIRE model.
The primary function of Miles is a requirement intended by designers. On the other hand, the secondary function is considered as a requirement also intended by the designers, but to a lesser degree. Both the primary and secondary functions pertain to a system; the primary function is expressed in terms of an effect on the environment and the secondary function on the system. Both these types of function resemble action (pump water, reduce noise, etc.) of the SAPPhIRE model in terms of the representation and level of abstraction.
Gero treats function as requirements intended by designers. This function pertains to the system but is described as an effect on environment. This function is similar to action of the SAPPhIRE model in the senses of representation and degree of abstraction.
Keuneke's functions are considered as requirements intended by designers. The states specified in the behavior can pertain to as effects on both, the system and its environment. But, since these functions satisfy the states, they can be considered as effect on system and environment. Keuneke's functions and states are similar to actions and state changes of the SAPPhIRE model in terms of representation and level of abstraction.
Ullman defines function as the transformation of inputs to outputs. Further, his examples seem to suggest that an abstraction of this transformation is also considered as function. The inputs, outputs, transformation and the abstraction of the transformation are requirements intended by designers. The inputs and outputs resemble the input of the SAPPhIRE model in terms of their representation as material, energy and signal. The abstraction of the transformation is similar to the action of the SAPPhIRE model. The transformation and its abstraction pertain to the system because both these occur inside the system boundary. The system interacts with its environment through inputs and outputs. Ullman also divides the overall abstraction of the transformation into sub-functions. This suggests the presence of sub-systems and elements at the functional level. These sub-functions can be seen as solutions for the overall function but can also be seen as requirements for developing concepts.
Pahl and Beitz's overall function is a requirement intended by designers. Since the sub-functions together satisfy the overall function, these are treated as solutions intended by the designers for the overall function. The sub-functions that can be further sub-divided are sub-systems at the function level; sub-functions that cannot be divided are elements at the function level. Both, the overall- and sub-functions are described within the system boundary. This shows that both these types of function pertain to the system. The relationship between system and environment is described using the input and output, both of which transcend the system boundary. Sub-functions are related to each other in terms of temporal relationships. Both the overall function and sub-functions are similar to action (measure liquid quantity, indicate liquid quantity, receive signal, channel signal, etc.) of the SAPPhIRE model in terms of the representation. The input and output of Pahl and Beitz are similar to input of the SAPPhIRE model in terms of level of abstraction.

Umeda et al. consider functions as requirements. Since functions are interpreted from behaviors by humans, functions are considered as intended by designers or users. The decomposition of required functions into sub-functions shows the presence of elements and sub-systems at the function level. The sub-functions together satisfy the required functions, and so these are considered as solutions. The functions pertain to a system but describe the effect of the system on the environment. The sub-functions pertain and describe effects within the system. The relationships between sub-functions are causal and task-based. The functions and sub-functions of Umeda et al. are similar to action of the SAPPhIRE model in terms of the representation and level of abstraction.

Simon also considers function as an intended requirement from designers. His function is similar to action in the SAPPhIRE model. In his function, he only uses information about the system but feels that any artifact is designed to perform only in particular environment(s) which signifies the relationship between system and environment.

Chakrabarti treats one view of function as an actual requirement with the same abstraction as behavior. This view of function is envisioned when the structure of the system operates in a way leading to intended or unintended functions by users. This level of abstraction uses information of parts (door handle, wedge, etc.) and phenomenon (press, retract, etc.) of the SAPPhIRE model. Since this view of function uses each process within the system, it signifies the presence of elements and sub-systems at the abstraction levels of phenomenon and part. Chakrabarti treats the other view of function as requirement intended by designers. This view of function pertains to the overall system and uses information about environment (door, wind, etc.). This view of function is similar to action (lock against wind) of the SAPPhIRE model because it is a higher level abstraction of behavior (phenomenon).

The function of Chittaro and Kumar pertains to components in a system. This is equivalent to functions of sub-systems and elements. Since functions are roles of components, they are considered as requirements. This function (barricade, conduct, etc.) is similar to phenomenon of the SAPPhIRE model. The teleology describes the intention of users or designers.

The environment-centric function of Chandrasekaran and Josephson pertains to the system and uses information of the environment. The device-centric function also pertains to the system and uses information of both, the system and the environment. The device-centric function uses information about the sub-systems (box) and elements (switch) in the system. Both these views are requirements intended by designers. Both these views are at different levels of abstraction; the environment-centric view being at a higher level of abstraction than the device-centric view. The environment- and device-centric views of function use information related to action (make sound) and, action (make sound) and parts (box, switch, etc.) of the SAPPhIRE model, respectively. Mode of deployment describes the intention of designers and users, and is expressed in terms of the relationship between system and environment at a level of abstraction similar to parts of the SAPPhIRE model.

The product function of Stone and Wood pertains to the overall system and is taken as a requirement intended by designers. The description of this function (transmit torque, convert electricity) is similar to action of the SAPPhIRE model. The sub-functions which together satisfy the overall product function pertain to sub-systems and elements, and are considered as solutions for the overall product function. Sub-functions as illustrated in Functional Basis are similar to action, state change, input and phenomenon of the SAPPhIRE model.

Hubka and Eder consider functions as intended requirements described as a transformation of intended primary and secondary inputs into intended primary and secondary outputs. The primary and secondary functions here pertain to system because functions are taken as the transformation of inputs that enter the system and outputs that leave the system. The system-environment relationship is explained using the primary/secondary inputs and outputs; all of these transit the system boundary. The input and output are similar to the input of the SAPPhIRE model in terms of its various kinds i.e., material, energy and information. The transformation of inputs to outputs is similar to state change of the SAPPhIRE model. The transformation of primary inputs to primary outputs is intended by designers but the transformation of secondary inputs to secondary outputs is unintended.

The purpose function of Deng involves human interpretation and is therefore, considered as a requirement intended by users. It is similar to action of the SAPPhIRE model in terms of level of abstraction and representation. This function pertains to the system. The action function is also taken as requirement intended by designers, and contains more information about the system in terms of its sub-systems and elements (rotate, constant angular velocity, hour hand, minute hand, etc.). The action function uses information similar to phenomenon (rotate), constant angular velocity (organ) and part (hour hand, minute hand, etc.) of the SAPPhIRE model. Since the action function involves interactions within the system and, between system and environment, there exist relationships between system and environment, within sub-systems and elements, and between sub-systems and elements. The action function can be seen as a solution for the purpose function.

The function in the SBF-model is taken as a requirement intended by designers pertaining to the system. This function which is a transition of an input to output, is similar to the state change of the SAPPhIRE model. At the behavior level, the overall transition from input to output is broken down into a series of state transitions. This shows breaking down of the overall function into sub-tasks which in turn can be seen as the input-output transitions of the sub-systems and elements of the system.

Kitamura and Mizoguchi's functions describe the different views of functions. The actual function refers to the function of the system and resembles the function at a higher level of abstraction. The capacity function resembles function at lower level of function. Since Kitmaura and Mizoguchi do not use any examples to illustrate these functions, we could not show the similarity between the levels of abstraction of these functions with the SAPPhIRE model. The artifact function is the function intended by the users for the system. The device function is the function intended by the users for the sub-systems or elements, which together constitute the system. The essential function is a requirement intended by the designers for the system. The accidental function is a function that is not intended by the designers for the system. Functions unintended from the designers can mean intended or unintended by the users.

The above explanation in this section is summarized in a table in [http://tinyurl.com/6qwarwd].

6. Discussion

The earlier section shows that all the functions reviewed in this paper can be described using the framework. The table also shows that by describing functions in terms of a common set of views provides a better means of comparing and assessing relationships between functions. In terms of the level of abstraction, majority of the functions described in literature are similar to the action description of the SAPPhIRE model. State change-, input-, phenomenon- organ- and part- descriptions are also used to describe functions. In terms of the requirement-solution view, most of the functions are described as requirements. Some functions are also treated as solutions. The literature reports only the functions of systems, sub-systems and elements. Since, environment is an important entity in designing for various reasons explained by Ranjan et al. [2012], the following questions arise. What is the function of the environment? What is its role in designing? The literature only reports functions that are intended or unintended from a designer's point of view, or, intended from a user's point of view. Functions of products that are unintended by designers can be either intended or unintended from a user's point of view. Functions of designs that are unintended from the designer's point of view but intended from the user's point of view are referred to as affordances. Functions of designs that are unintended for both designers and users, are undesired side effects that might lead to accidents.

7. Conclusions

The literature on function reveals that there exist many views of function, and not all these views are made explicit. Based on the literature survey on functions, the following views were identified in this paper: level of abstraction, requirement-solution, system-environment and intended-unitended from users'/designers' point of view. Due to a strong resemblance between these views and earlier research, a framework has been developed using the SAPPhIRE model, requirement-solution model, system-environment model and intended-unintended from designers' and users' views. The different functions are described using the framework. A theoretical validation of the framework shows that the different views of function can be described using the framework. The framework also provides a common basis for comparing and assessing relationships among the functions.

References

Balachandran, M., Gero, J., "Role of prototypes in integrated expert systems and CAD systems", Applications of Artificial Intelligence in Engineering, Gero, J. (ed.), Vol. 1, Springer–Verlag Berlin, Germany, 1990.

Brown, D. and Blessing, L., "The relationship between function and affordance", Proceedings of IDETC/CIE, California, USA, 2005, DECT2005-85017. (CD-Proceedings).

Chakrabarti, A., "Supporting two views of function in mechanical design", Proc. Workshop on Functional Modelling and Teleological Reasoning, 15th AAAI National Conference on AI, Wisconsin, USA, 1998.

Chakrabarti, A., Blessing, L. "Special issue: representing functionality in design", Guest Editorial, AI EDAM, 10(4), 1996, 251-253.

Chakrabarti, A., Sarkar, P., Leelavathamma, B., Nataraju, B., "A functional representation for aiding biomimetic and artificial inspiration of new ideas", AI EDAM, 19(2), 2005, 113–132.

Chandrasekaran, B., Josephson, J. "Function in device representation", Engineering with computers, 16, 2000, 162–177.

Chittaro, L., Kumar, A., "Reasoning about function and its applications to engineering", AI in Engineering, 12(4), 1998, 331-336.

Deng, Y., "Function and behavior representation in conceptual mechanical design", AI EDAM, 16(5), 2002, 343–362.

Deng, Y., Britton, G., Tor, S., "Consraint-based functional design verification for conceptual design", Computer-Aided Design, 32(14), 2000, 889–899.

Gero, J., "Design prototypes: A knowledge representation schema for design", AI Magazine, 11(4), 1990, 26–36.

Goel, A., Rugaber, S., Vattam, S., "Structure, behavior, and function of complex systems: the structure, behavior, and function modeling language", AI EDAM, 23(1), 2009, 23-35.

http://tinyurl.com/6qwarwd, 2012.

Hubka, V., Eder, W., "Functions revisited", Proc. International Conference on Engineering Design, ICED 01, Glasgow, UK, 2001. (CD Proceedings).

Hubka, V., Eder, W., "Theory of technical systems and engineering design synthesis", Engineering Design Synthesis, Chakrabarti, A. (ed.), Springer-Verlag London, UK, 2002.

Keuneke, A., "Device representation - the significance of functional knowledge", IEEE Expert, 6(2), 1991, 22–25.

Kitamura, Y., Mizoguchi, R., "Characterizing functions based on ontological models from an engineering point of view", Proc. Formal Ontology in Information Systems (FOIS), IOS Press, Amsterdam, 2010, 301-314.

Maier, J., Fadel, G., "Affordance-Based Methods for Design", Proc. DTM IDETC/CIE, Chicago, IL, 2003.

Miles, L., "Techniques of Value Analysis and Engineering", McGraw–Hill, New York, USA, 1972.

Pahl, G., Beitz, W., "Engineering Design: A Systematic Approach", Springer–Verlag, Berlin, Germany, 1996.

Ranjan, B., Srinivasan, V., Chakrabarti, A., "System-Environment View in Designing", CIRP Design, Bangalore, India, 2012. (accepted)

Rodenacker, W., "Methodisches Konstruieren", Springer– Verlag, Berlin, Germany, 1971.

Simon, H., "The sciences of the artificial", MIT Press, Boston, USA, 1996.

Srinivasan, V., Chakrabarti, A., "An integrated model of designing", JCISE, 10(3), 2010, 031013.

Srinivasan, V., Chakrabarti, A., "SAPPhIRE: An approach to analysis and synthesis", Proc. International Conference on Engineering Design, ICED 09, Stanford, USA, 2009. (CD-Proceedings)

Stone, R., Chakrabarti, A., "Special issue: Engineering applications of representations of function", Guest editorial, AI EDAM, 19(2), 2005, 63.

Stone, R., Wood, K., "Development of a functional basis for design", J. Mech. Des., 122(4), 2000, 359-370.

Ullman, D., "The Mechanical Design Process", McGraw-Hill, Boston, USA, 1992.

Umeda, Y., Ishii, M., Yoshioka, M., Shimomura, Y., Tomiyama, T., "Supporting conceptual design based on the function-behavior-state modeller", AI EDAM, 10(4), 1996, 275–288.

Vermaas, P., "Accepting ambiguity of engineering functional descriptions", Proc. International Conference on Engineering Design (ICED11), Copenhagen, Denmark, 2011. (CD-proceedings)

Vermaas, P., "The flexible meaning of function in engineering", Proc. International Conference on Engineering Design, ICED 09, Stanford, USA, 2009. (CD Proceedings).

Wood, K., Greer, J., "Function-based synthesis methods in engineering design," Formal Engineering Design Synthesis (Ed.: Antonsson, E and Cagan, J.), Cambridge University Press, USA, 2001.

Dr. Srinivasan V
Research Assistant
Technical University of Munich, Institute of Product Development
Boltzmannstr. 15, 85748 Garching, Germany
Telephone: 089 289 15137
Telefax: 089 289 15144
Email: srinivasan.venkataraman@pe.mw.tum.de
URL: http:// www.pe.mw.tum.de