

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/243776423>

Theoretical Underpinnings of Functional Modeling: Preliminary Experimental Studies

Conference Paper · September 2000

DOI: 10.1115/DETC2000/DTM-14563

CITATIONS

38

READS

12,836

5 authors, including:



[Kristin Lee Wood](#)

University of Colorado

654 PUBLICATIONS 20,423 CITATIONS

SEE PROFILE



[Kevin Norbert Otto](#)

University of Melbourne

234 PUBLICATIONS 7,701 CITATIONS

SEE PROFILE

DETC2000/DTM-14563

THEORETICAL UNDERPINNINGS OF FUNCTIONAL MODELING: PRELIMINARY EXPERIMENTAL STUDIES

Mark A. Kurfman & Robert B. Stone[†]

Mechanical Engineering & Basic Engineering Departments
University of Missouri-Rolla
Rolla, MO 65409-0210
rstone@umr.edu

Mike Van Wie & Kristin L. Wood

Department of Mechanical Engineering
The University of Texas at Austin
Austin, TX 78712
wood@mail.utexas.edu

Kevin N. Otto

Center for Innovation in Product Development
Massachusetts Institute of Technology
Cambridge, MA 02139
knotto@mit.edu

ABSTRACT

A model of how a product should function to satisfy customers is an essential element in clarifying, identifying, and establishing product architectures. Such functional models greatly enhance the generation of creative form solutions to a chosen architecture. A wider breadth of solutions is generally possible, implementing new and stable technologies. In turn, using the recent concepts of design repositories, the possibilities exist to archive, retrieve, compute, reconfigure, and reason with the product forms. To realize these benefits to the fullest extent possible, functional modeling needs further theoretical development. A formalism of function classes, vocabulary, topologies, and methodology is a first step towards this goal. Recent research efforts have focused on each of these elements, where great strides toward repeatable formalisms have been made. Yet, across the engineering design field, very little active experimentation has been pursued to test the veracity of these elements, individually and as a whole. We address this issue here through a preliminary set of experiments conducted at three separate universities. Design teams and individuals are asked to create functional models, in the context of product development, with and without the formalisms. The outcomes of the modeling effort are analyzed to determine the repeatability of the process. Early results are quite encouraging. Very repeatable results are obtained for three product evolutions, including a toaster, a power screwdriver, and a toy dart gun. In addition, weaknesses in current formalisms are un-

covered, pointing to new directions for advancing the field and for carrying out more advanced experimentation.

1. INTRODUCTION

Functional modeling is a key step in the product design process, whether original or redesign. This article reports on our collaborative research efforts to develop a theoretical approach to functional modeling. By developing a formal theory of functional modeling, we intend to push functional modeling into the realm of repeatable, and even computable, engineering analysis. While we have not yet reached our final destination on this journey, substantial progress has been made with our functional model derivation and common functional language as demonstrated by inter-institutional experimental results. Namely, we show that our flow tracing methodology to function structure development provides reasonably repeatable results in function structure form, across a variety of persons trained in the approach.

Several factors motivate the research into the theoretical underpinnings of functional modeling. In particular, use of the functional modeling theory described in this article significantly contributes to the following six product design areas.

- *Systematic function structure generation.* The most common criticism of functional models (particularly their graphical representation known as a function structure) is that a given product does not have a unique representation. Even within a systematic function-based design methodology, different designers can produce differing function structures, even under the same process choices. A common set of func-

[†] Corresponding author: 102A Basic Engineering Building, University of Missouri-Rolla, Rolla, MO 65409-0210; Phone: 573-341-4086; E-mail: rstone@umr.edu

tions and flows, part of the groundwork for this theoretical development, significantly reduces this occurrence. However, a first-principles-type approach is needed to uncover the true functionality of a product.

- *Design education.* Functional modeling provides a consistent basis for developing high-level physical models, and for teaching the abstract concepts of functional modeling to engineers.
- *Product architecture development.* The desire to define the product architecture of a product as early as possible in the conceptual design stage necessitates basing the decision on a functional model of the product (Stone *et al.*, 1998 & 1999). Functional clumping rules or heuristics can be applied to identify product modules, leading to clear definitions of modular interface requirements. Similar heuristics have also been developed to identify modules for use across a product family (Zamirowski and Otto, 1999). In general, the approach leads to alternative layouts where concept generation techniques may then be used to embody the layouts and sub-modules. To do this systematically across a wide variety of products, repeatable functional modeling techniques are needed.
- *Design by analogy.* Few product designs are truly “original.” Instead, they incorporate elements of other product designs that have accumulated in the corporate body of design knowledge. If functional descriptions of products, expressed in a common language and causality, are represented and archived in a repository, then the repository can be searched to find products similar in function and architecture. This offers obvious applications to benchmarking products and searching for form solutions.
- *Creativity in concept generation.* The ability to decompose a design task is fundamental to arriving at creative solutions (Ullman, 1997). Likewise, it is critical to represent abstract and incomplete information to make decisions early in a design process or product development. Functional models, with the addition of a functional basis, significantly aid the capacity of design teams to break problems down and make critical early decisions.
- *Product metrics, robustness, and benchmarks.* An important aspect of product development is to formulate objective measures for benchmarking and quality endeavors. Functional models can greatly enhance methods, such as Quality Function Deployment, in identifying and choosing metrics. The flows of functional models provide a high-level physical model of a product’s technical process. These flows, if suitably formalized, are directly measurable, reducing the guesswork and artistic nature of choosing metrics.

In this article, we look only at the functional modeling portion of the conceptual design process. However, the theoretical underpinnings of functional modeling developed here are applicable to any number of specific overall design methodologies. In Section 2 we review the design research in the functional modeling area. Section 3 contains our theoretical development of the

functional modeling derivation along with the theory behind the topology and product architecture tools used for later evaluation of the theory. Analysis of our collaborative experiments designed to test the functional model derivation is presented in Section 4. Section 5 lays out our ongoing work in the area, and we assess the impact of a functional modeling theory in Section 6.

2. BACKGROUND

Function-based engineering design begins with the establishment of an overall product function. This overall product function is then broken down into sub-functions, and eventually these sub-functions are used to develop the form of the product. Currently there are many different function-based design methodologies (Pahl and Beitz, 1996; Ullman, 1997; Otto and Wood, 2000; Hubka, 1984; Ulrich and Eppinger, 1995; Schmidt and Cagan, 1995; Pimpler and Eppinger, 1994; Shimomura *et al.*, 1996; Cutherell, 1996). Their usefulness in the design process is well established, but their limitations are also well known. One area for concern with these design methodologies is the possibility to produce dissimilar functional models under agreed-upon process choices and assumptions. When asked to develop a functional model, two engineers given the same product, customer needs, and process descriptions will possibly create two completely different functional models. One possible explanation for this inconsistency is the lack of a common vocabulary and systematic approach to functional model development. This is not a new revelation in function-based design and has received the attention of design theory researchers in recent years.

There have been many attempts to create a common language, and to a lesser extent, a theory of functional modeling, for design. Value analysis is one of the earliest such examples as it seeks to express the sub-functions of a product as an action verb-object pair and assign a fraction of a product’s cost to each sub-function, formed in a hierarchical tree structure (Miles, 1972; Akiyama, 1991; VAI, 1993). Suggested verb-object lists are given by Miles and VAI for different disciplines; however, no overall language is proposed. Collins *et al.* (1976) developed a list of 105 mechanical functions in order to classify helicopter failure information. It is a valuable system, but limited since it only deals with helicopter systems. Other works, such as the Theory of Inventive Problem Solving (Altshuller, 1984; Malmqvist *et al.*, 1996), living systems theory (Koch *et al.*, 1994), and the Pahl and Beitz-inspired works (Kirschman & Fadel, 1998; Hundal, 1990; Pahl & Beitz, 1996; Hubka, 1984; Murdock *et al.*, 1997; Lai & Wilson, 1989; Iwasaki *et al.*, 1995; Umeda & Tomiyama, 1997) have made significant contributions to functional modeling approaches as well. Another approach attempting to standardize a common functional vocabulary is by Szykman *et al.* (1999). They have proposed a standardized set of functions and flows as part of a computable data structure to represent product function and its link to product form. This mild form dependence presents an obstacle to its use during conceptual design where the creativity of product concepts often relies on divorcing function from form.

Our previous research in functional modeling indicates that a theoretical underpinning for functional modeling is needed for functional modeling to emerge as a repeatable and insightful analysis tool (Otto, 1996, Stone et al., 1999 & 2000; Otto & Wood, 2000; Stone & Wood, 1999). We introduce our theoretical statement of functional modeling and detail several preliminary experiments, conducted at the academic level, to assess the theory's impact in design.

3. THEORETICAL UNDERPINNINGS OF FUNCTIONAL MODELING

Over many years of doing design and teaching design, we have found more and more benefits from developing a good functional model of the product. Our methods for generating functional models have matured as well. These intertwining methods and techniques are woven together here to form the theoretical underpinnings of functional modeling. This theory will no doubt continue to evolve, but represents the first comprehensive statement that we are aware of which links customer needs to functional models and is highly repeatable. Our overall theoretical approach is to develop a repeatable functional model derivation methodology that builds a foundation for techniques that aid a designer in synthesis, in addition to a machine interpretable model. We begin with the functional model derivation methodology, first presented by Otto (1996) without the common basis development.

3.1 DERIVING A FUNCTIONAL MODEL

TASK 1: IDENTIFY FLOWS THAT ADDRESS CUSTOMER NEEDS

The first task of the functional model derivation is to identify the flows (the physical phenomena) that the product is to operate on. We initiate this process by identifying and listing the flow or set of flows that addresses each customer need. Therefore, a complete set of customer needs for the product is needed. Systematic and repeatable techniques for gathering customer needs are well described in the literature (Urban and Hauser, 1993; Otto and Wood, 1997; Ullman, 1997; Ulrich and Eppinger, 1995). In general, we find customer needs primarily identify input/output flows for the product, not flows internal to the product. If a customer need does not relate directly to a flow, then the customer need is most likely a constraint. A constraint is not something the product does (hence no associated flows), but rather is a holistic property of the product. For example, the customer need for a product to be *low cost* does not relate to a specific flow of energy, materials, or signals. Instead, it is a holistic property (each element in the product contributes) and it imposes a constraint on the product design.

To demonstrate Task 1 of the functional model derivation, consider the customer need to flow correlation for a redesign of a Black and Decker DustBuster, shown in Fig. 1. The related flows are identified for the customer needs and are shown in Table 1.



Figure 1. The DustBuster portable vacuum used in the functional model derivation example.

Table 1. Task 1: Identification of the flows that address customer needs for a redesign of a Black and Decker DustBuster.

Customer Need	Importance rating (1 to 5)	Related flows
Provide cord	2	elect. ener.
Provide attachments	2	constraint
Easy storage	1	constraint
Stronger / easier mounting	2	human ener.
Easy to empty	2	debris, hand
Durable filter / bag	3	debris, air
More power	3	elect. ener.
More suction	3	pneumatic ener., air
Lightweight	3	constraint
Well balanced	4	weight
Long battery life	3	elect. ener.
Fast recharging	3	elect. ener.
Indicator light	3	visual signal
Smaller size	1	constraint
Shape for corners	3	constraint
Comfortable handle	2	hand
Different color	2	constraint
Sturdier case	5	impact force
Use frequently	1	elect. ener.
Wet / dry capabilities	4	debris, air
Easy to operate	1	human ener., on / off

TASK 2: GENERATE A BLACK BOX MODEL

Once a list of flows for each customer need is complete, we create a black box model, a graphical representation of product function with input/output flows. The overall function of the product is expressed in verb-object form. The input/output flows should be drawn from the flow listing of Task 1. The Black Box model represents a high-level transfer function that is based on customer needs. The Black Box model for the DustBuster is shown below in Fig. 2.

Table 2. Flow classes and their basic categorizations.

Class	Material	Signal	Energy		
Basic	Human	Status	Human	Electrical	Mechanical
	Gas	Signal	Acoustic	Electromagnetic	Pneumatic
	Liquid		Biological	Hydraulic	Radioactive
	Solid		Chemical	Magnetic	Thermal

Table 3. Function classes and their basic categorizations.

Class	Basic	Class	Basic	Class	Basic
Branch	Separate	Control Magnitude	Actuate	Signal	Sense
	Refine		Regulate		Indicate
	Distribute		Change		Display
	Dissipate		Form		Measure
Channel	Import	Convert	Convert	Support	Stop
	Export	Provision	Store		Stabilize
	Transfer		Supply		Secure
	Guide		Extract		Position
Connect	Couple				
	Mix				

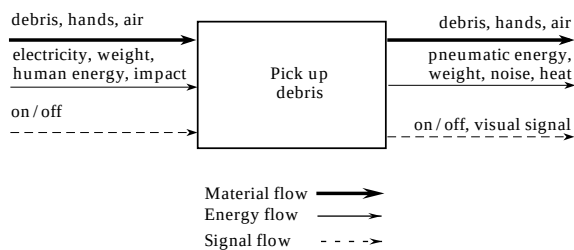


Figure 2. The Black Box model for the DustBuster.

TASK 3: CREATE FUNCTION CHAINS FOR EACH INPUT FLOW - “BE THE FLOW”

For each input flow from Task 2, this task develops a chain of sub-functions that operate on the flow. Here, the designer must ‘become the flow.’ As the flow, the designer imagines each operation on the flow from entrance until exit of the product (or transformation to another flow) and expresses it as a sub-function in verb-object form. If a flow is transformed to another type, then we follow the operations on the transformed flow until it exits the product.

Subtask 3A: Express sub-functions in a common functional basis

The function chains (and the subsequent functional model) are translated into the standard vocabulary of the functional basis, shown in Tables 2 and 3. That is, all product specific flows (crumbs, 12V, etc.) are abstracted into one of 8 generic flows (Table 2), and all product specific functions (catch crumbs, convert AC to DC, etc.) are abstracted into one of 26 generic functions (Table 3). Selecting functions and flows (from Tables 1 & 2) from the standard vocabulary are combined in verb-object form to describe a sub-function. Expressing a functional model in functional basis form provides the general benefit of comparable function structures among different designers – all are expressed using the same syntax. Furthermore it offers a standard level of detail for functional models and a means of verifying the consistency and cor-

rectness of the physical system, and an important stepping stone for education.

Subtask 3B: Order function chains with respect to causality

Next, the functional model is ordered with respect to causality. In *sequential function chains* a causal link exists between sub-functions, i.e. they must be performed in a specific order to generate the desired result. A flow common to all these functions is termed a *sequential flow*.

Parallel function chains consist of sets of *sequential function chains* sharing one or more common flows. Graphically, they are represented by a flow that branches in a functional model. Collectively, the chains are called *parallel* because they all depend on a common sub-function and flow, but are independent of each other. Independence means that any one of the chains of the parallel function chain set does not require input from any other chain within the set. Physically, the parallel function chains represent different components of a device that may operate all at once or individually. Two example function chains for the DustBuster are shown in Fig. 3 after being transformed into the functional basis language.

TASK 4: AGGREGATE FUNCTION CHAINS INTO A FUNCTIONAL MODEL

The fourth task of functional model derivation is to aggregate all of the function chains from Task 2 into a single model. It may be necessary to connect the distinct chains together. This action may require the addition of new sub-functions or their combination, defining the interfaces of modules within the representation. The functional model for the DustBuster, after aggregation of its function chains, is shown in Fig. 4.

TASK 5: VERIFY THE FUNCTIONAL MODEL WITH CUSTOMER NEEDS

As a final check of the functional model, the next step entails checking that each customer need (which is not a constraint) is

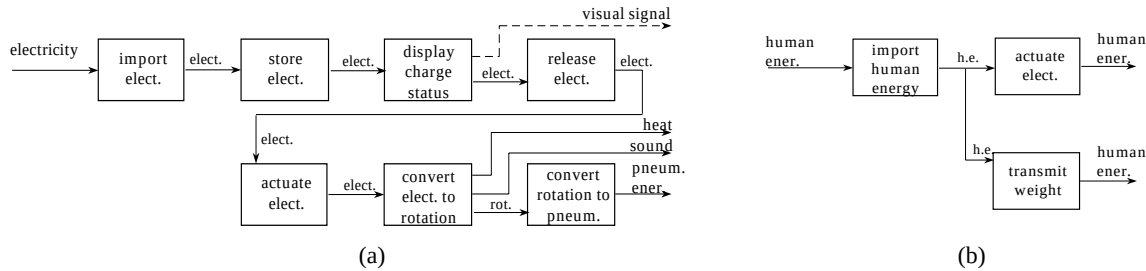


Figure 3. Function chains for the DustBuster expressed in terms of the functional basis. (a) A sequential function chain formed by following the flow of electricity through the product. (b) A parallel function chain formed by the flow human energy.

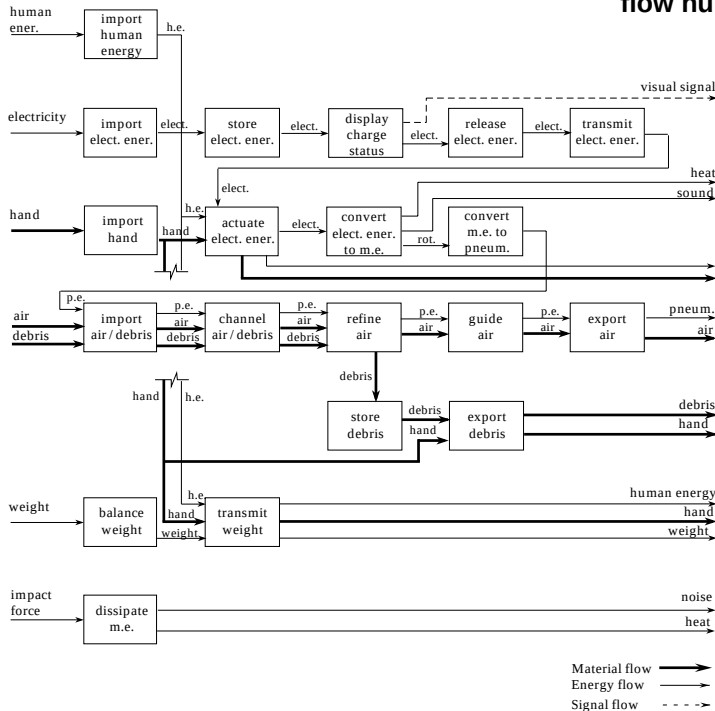


Figure 4. The complete functional model for the DustBuster after aggregating the function chains.

addressed by at least one sub-function. If any customer needs remain unmet, we iterate through the tasks again, starting at Task 2.

The result of the derivation is a functional model of a product that is expressed in the functional basis language and that is repeatable, given the same customer needs and process choices. With such a functional model, functions may be directly related to customer needs, products and their functional representations may be directly compared, product families may be identified, product functions may be prioritized, and direct component analogies may be generated within and outside product classes.

3.2 FUNCTIONAL TOPOLOGY - A STRUCTURE INDEPENDENT EXPRESSION OF FUNCTION

One of the major benefits to the functional derivation method of the previous section is the ability to create repeatable functional models given specific process choices. While the resulting func-

tion model provides an excellent graphical representation of product function, it is difficult to test the equivalence of several functional models. Additionally, if functional modeling is to become a computable (machine interpretable) technique, then a numerical representation that captures flow information as well as function is needed.

We introduce here the *function topology adjacency matrix*, an alternative representation that incorporates function and flow information in a computable form. The adjacency matrix represents the *functional topology* of a product. Functional topology is defined as the ordered and connected arrangement of functions and flows for a set of customer needs. Based on this definition, the concept of functional equivalence is defined. Two functional models are equivalent if their number of sub-functions and their functional topologies are unaffected by spatial distortions.

A functional topology consists of three informational items: 1) a set of sub-functional descriptions, 2) a set of flows (input, output and intermediate) and 3) the ordered connections of the sub-functions by the flows. A very simple, and abstract, example of the functional equivalence of two functional models is shown in Fig. 5. The two functional models have a different spatial layout, but are topologically the same as evidenced by the adjacency matrix they share. The adjacency matrix is read by entering along a row and reading over to a column. The row label is the location from which the flow originates and the column label represents the location to which it proceeds. The numerical code that symbolizes a flow is entered in the appropriate matrix cell to indicate a connection. The adjacency matrix is divided into four quadrants (separated by double lines in Fig. 5). The upper left quadrant represents any input flow to sub-function connections. The lower left quadrant shows flow connections between sub-functions. The lower right quadrant traces the flows' route from their last sub-function as they cross the system boundary. The final upper right quadrant will always be null as it would represent flows which directly pass through a product without any sub-function operation.

The topology of different functional models can be compared directly by combining adjacency matrices into a *frequency adjacency matrix*, showing the frequency with which flow connections occur. Furthermore, with these two theoretical developments – the functional model derivation method and the func-

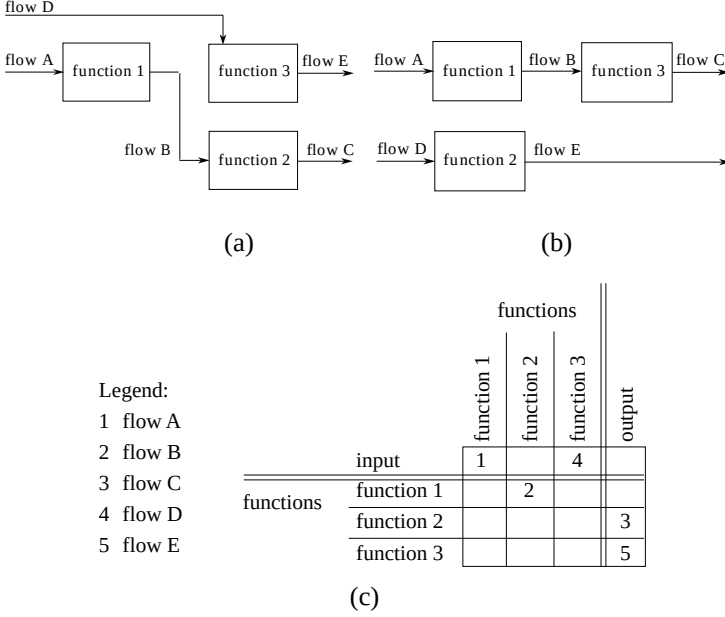


Figure 5. Two example functional models in (a) and (b) are spatially different, though they are shown to be topologically equivalent in (c).

tional topology concept – we can test our claim that functional models can be generated in a repeatable manner.

3.3 PRODUCT ARCHITECTURE THEORY

With a functional model expressed in the common language of the functional basis, sub-functions can be clustered to define modular product architectures. Stone, Wood and Crawford (1998) develop a set of three heuristics for identification of modules from a functional description. This work provides a systematic method for clustering sub-functions, building on previous approaches to modular product architecture. The heuristics require a functional model derived from the steps in Section 3.1, where sub-functions are then clustered based on flow (energy, material, or signal) relationships. The heuristics are stated below and shown schematically in Figure 6.

Dominant-Flow Heuristic: The set of sub-functions which a flow passes through, from entry or initiation of the flow in the system to exit from the system or conversion of the flow within the system, define a module.

Branching Flow Heuristic: Parallel function chains associated with a flow that branches constitute modules. Each of the modules interfaces with the remainder of the product through the flow at the branch location.

Convert-Transmit Heuristic: A conversion sub-function or a conversion-transmission pair or proper chain of sub-functions constitutes a module.

Zamirowski and Otto (1999) present additional similar heuristics for portfolio architecture over multiple products. We restrict here to single product architecture, though our arguments and methods directly apply to the expanded product family

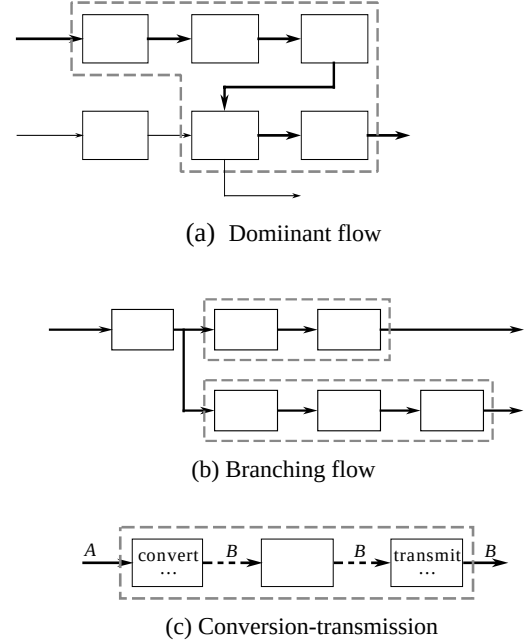


Figure 6. Schematic representation of the three heuristics for product architecture definition.

modularization problem well. Application of the three heuristics above generates a set of possible modules for a product. The modules can then be used to guide the concept development phase and the product embodiment.

3.4 PRODUCT-FUNCTION REPOSITORIES

The previous sections develop the theoretical underpinnings to create functional models and their associated product architectures. In this section the theory is extended to represent, analyze and archive sets of products. The concept of a product-function matrix is for this purpose.

A product function matrix is composed of a vector (column) for each product where its elements (rows) are weighted customer-need importance values for each of its functions. The product vectors are arranged into a $m \times n$ product-function matrix, Φ (McAdams et al., 1998; Stone et al., 1999a & b). Each element is the cumulative customer need rating for the i th function of the j th product. We must normalize this matrix to take into account differences in the number of customer needs and functions for each product, to remove biases for any one product.

Once implemented, the normalized version of Φ , N , has elements

$$v_{ij} = \phi_{ij} \left(\frac{\bar{\eta}}{\eta_j} \right) \cdot \left(\frac{\mu_j}{\bar{\mu}} \right), \quad (1)$$

where the average customer need rating is

$$\bar{\eta} = \frac{1}{n} \sum_{i=1}^m \sum_{j=1}^n \phi_{ij}, \quad (2)$$

the total customer need rating for the j th product is

$$\eta_j = \sum_{i=1}^m \phi_{ij}, \quad (3)$$

the number of functions in the j th product is

$$\mu_j = \sum_{i=1}^m H(\phi_{ij}), \quad (4)$$

and the average number of functions is

$$\bar{\mu} = \frac{1}{n} \sum_{i=1}^m \sum_{j=1}^n H(\phi_{ij}), \quad (5)$$

where H is a Heaviside function, n is the number of products and m is the total number of different sub-functions for all products. Normalizing the matrix provides a level playing field on which to compare products. The averaging and scaling mathematics defined above are an intuitive way to account for variations in customer needs and functional models. We may now use this representation to determine the critical functions across the domain or any sub-domain of products. For example, by post multiplying the product-function matrix by its transpose, we will have a function-function matrix, where the elements represent the importance of a single function or of any pairs of functions together. Alternatively, we can pre-multiply by the transpose to result in a product-product matrix where the elements represent the similarity or commonality of important functions that products share.

4 TESTING AND RESULTS

Using these theoretical underpinnings, we may now test our hypothesis that the functional model derivation method can result in repeatable functional models among different designers. We have conducted several experiments with students at our respective universities. The experiments focus on iterative functional modeling. The experimental procedures are described below along with the generally positive findings and recognized weaknesses.

4.1 ITERATIVE FUNCTIONAL MODELING EXPERIMENTS

Between the three universities, UT-Austin, MIT and UMR, three different functional modeling experiments were conducted. The first experiment, conducted in a UT-Austin and MIT graduate course, consisted of a 20 students each generating a functional model for a common slotted toaster. Students used their own toaster products for this analysis. The initial functional model was created following the general functional model derivation method, though the model was not converted to the functional basis language before aggregation. After the initial attempt at generating a functional model, the students were asked to transform their aggregated functional model into its functional basis equivalent.

The second and third experiments, conducted in a UMR undergraduate course, consisted of four students each generating a

functional model for a power screwdriver and a toy dart gun without using the functional basis. Students were provided with the same electric screwdriver and dart gun products. Once again, after the students generated the first functional model for each product, they were asked to transform their functional model into its functional basis equivalent.

4.2 ITERATIVE EXPERIMENT RESULTS

Methodology Evaluation

The overall effect of the functional model derivation methodology is evaluated first. In this section we are evaluating the functional models after conversion to the functional basis for the three experiments described in Section 4.1. For each product, the functional model derivation methodology generated repeatable results by the students as long as they started with consistent customer needs.

To illustrate the consistency of the models, three sample student functional models (representing a range of modeling skill levels) and the authors' experimental control functional model are shown in Figs.7 - 10. The observed consistency is three-faceted: 1) within function chains, the ordering of sub-functions is very repeatable (refer to Task 3B); 2) the use of sequential or parallel chains is consistent; and 3) the number of sub-functions used to describe the product is consistent, with a standard deviation of 3 on a scale of 0 to infinity.

Quantitative statements about the consistency of derived functional models can be made as well and are summarized in Table 4. Across all three products, 50% of the total set of sub-functions is identified by more than 70% of the students. Additionally, 70-100% of the sub-function set is identified by more than 50% of the students. These figures offer resounding support for the repeatability of the functional model derivation method, especially since this is the first time the students ever performed functional modeling.

Table 4. Percentages of the average sub-function set identified by percentages of students.

% of students	% of avg. sub-function set identified	
	with functional basis	without basis
50	70-100	20-80
70	50	0-20

As encouraging as these results are, there are some differences that remain between the functional models. We conclude that these differences are due to three reasons. First, different customer needs arise for different product families. Because the toaster experiment uses differing toaster products, the number of customer needs is different. However, these differences are not significant, as the core customer needs across all toasters are in fact equivalent. Second, there are errors in the functional models such as the use of a basis function contrary to its definition and the inappropriate causality of function chains contrary to the physi-

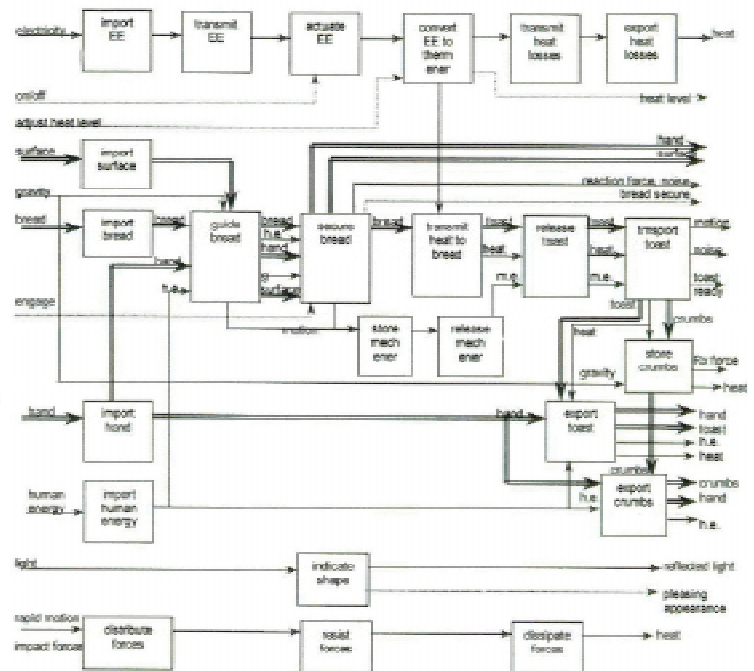
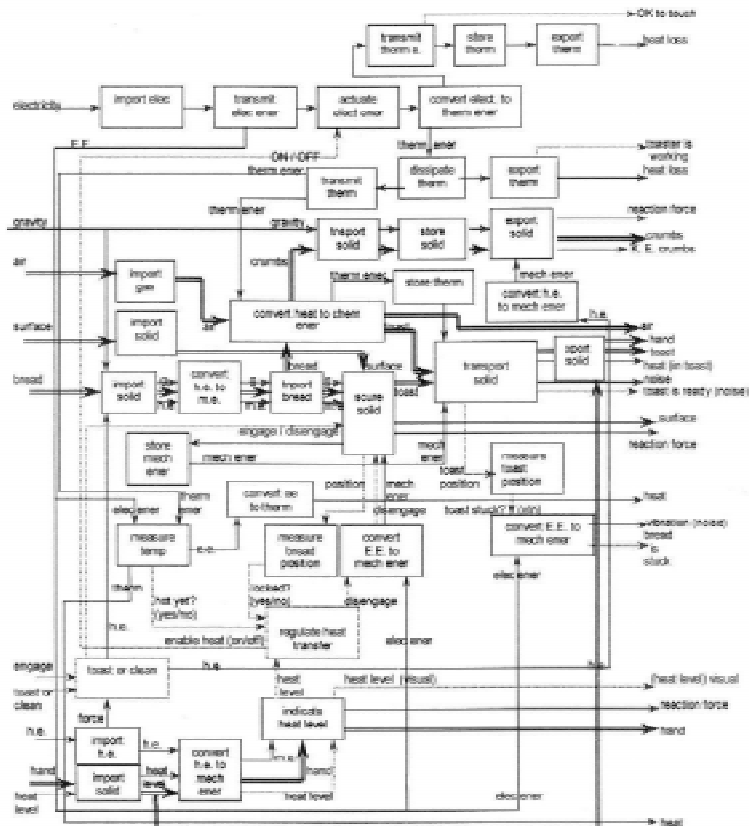


Figure 7. A toaster functional model where the transformation to the functional basis led to a deeper understanding, and simplification in this case, of the previous model.

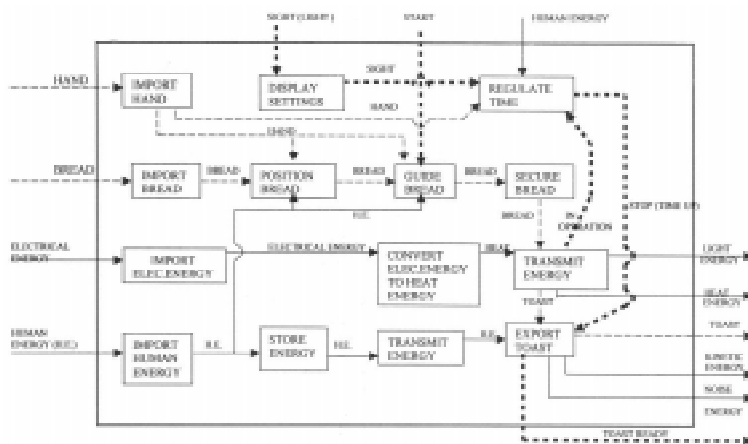


Figure 8. A toaster functional model at a typical level of modeling skill.

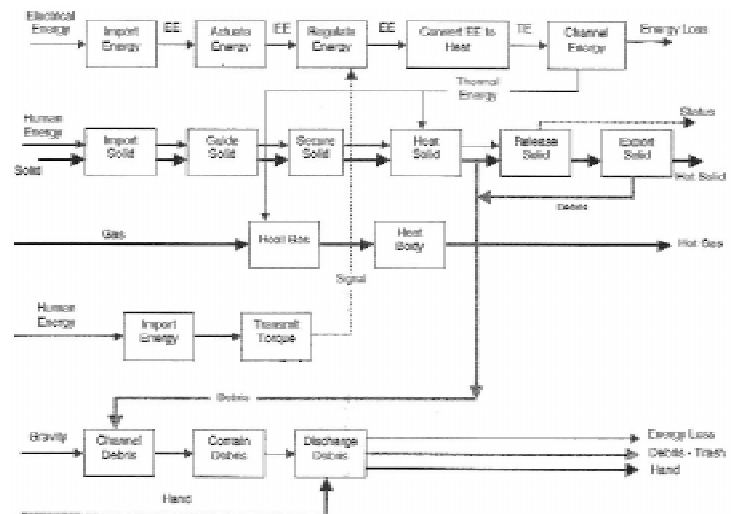


Figure 9 A toaster functional model representing a "good" level of modeling skill.

Table 5. Percent reduction in the size of the sub-function space with the functional basis.

	Functions	Generic Functions	Average Number of Sub-functions	% Reduction
Toaster	155	47	18	70
Power Screwdriver	47	28	15	40
Dart Gun	31	24	11	24

scriptions. After expressing the functional models in functional basis form, the size of the sub-function space drops by 70%, for an average number of sub-functions of 18 per model. Similar Results occur for the power screwdriver and dart gun.

This reduction in the size of the sub-function space in some cases reduces the complexity and improves the clarity of a function model (as it did in Fig. 7), but it may also do the reverse – it may uncover necessary functionality that was missing in the earlier version. The constraining of the sub-function space offers a marked improvement in the ability to communicate the product functionality and compare it with other products. This is particularly useful when searching for solution principles to sub-functions. For each of the products, the sub-function space for the functional basis form of the model and the individual sub-function frequency is shown in Figures 11 - 13. Additionally, a sample of the sub-function space for the non-basis toaster functional models is shown in Fig. 14, demonstrating the wide variability in sub-function descriptions. From these results, we observe that the percent reduction of sub-functions increases with the increasing average number of functions, as shown in Table 5.

Topological Evaluation

To evaluate the correctness of the functional topology (flow connections of the functional models), the functional models created using the basis are entered into an adjacency matrix. As discussed in earlier sections, an understanding of functional topology allows us to compare different functional models without the concern of possible spatial distortions.

For each product, all of the functional models were entered into adjacency matrices and the individual adjacency matrices were compiled into a frequency adjacency matrix. The toaster frequency adjacency matrix is included in Fig. 15. If every functional model is exactly the same, i.e. 100% repeatable, then the frequency of any non-zero cell would be 1. We are not there yet, but the frequency adjacency matrix does show significant repetition of topology. More encouraging is the clumping of flow connections indicating nearly identical sets of function chains. The high frequency connections displayed in Fig. 15 indicate that several sub-functions such as *import*, *export*, *convert* and *transfer* are well understood. There does appear to be some confusion, however, on the use, or at least the sequence, of functions such as *actuate*, *store*, *guide* and control-related functions.

In sum, this is an early look at functional topology. The adjacency matrix provides a framework for making functional models computable, whether for comparison purposes or other numerical manipulations. As expected, the frequency adjacency

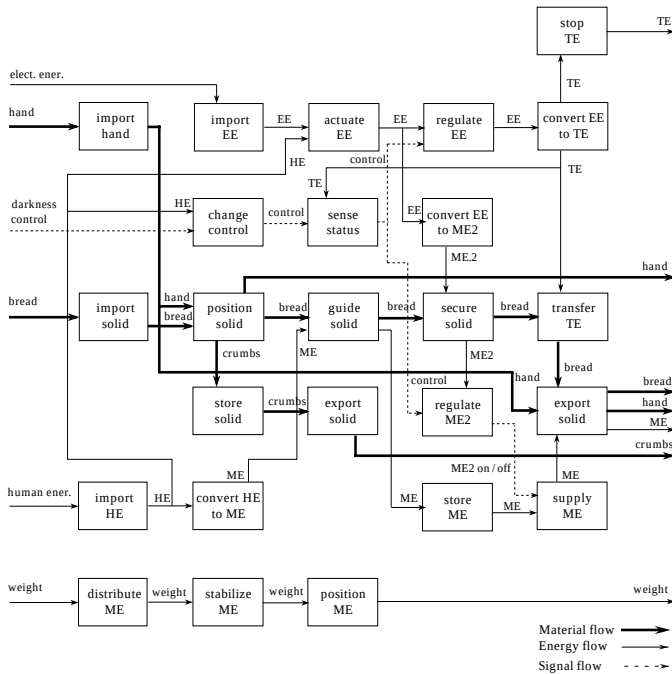


Figure 10. The experimental control functional model for the toaster.

cal process. Student inexperience with functional modeling is responsible for creating these types of mistakes. In the third case, the modeling errors are traceable to ill-defined functional basis definitions. For example, the control functions in the functional basis are stated at too high a level of abstraction to adequately describe the operation of real world controllers.

Each of these causes for modeling differences is addressed in this paper. Product family differences will be eliminated in future work as noted in Section 5. The *modeler* errors are assessed by using an experimental control functional model (such as Fig. 10 for the toaster product) developed by the authors for each of the products in the following sub-sections. Errors due to our functional basis definitions serve as feed back in our continuing development of the functional modeling theory and will be addressed in future work. We note that the definitions for control functions and flows need further refinement.

Overall Reduction in the Number of Sub-function Descriptions

In this section, we analyze the effect that the functional derivation methodology has on the size of the sub-function space to describe a product. In the initial modeling effort for each of the products, sub-functions are generated without the aid of the functional basis to constrain the sub-function space. The second modeling effort transforms the initial functional model and expresses it in the language of the functional basis. The size of the sub-function space for each modeling effort is computed and compared. These results are shown in Table 5.

For the toaster functional models generated without the functional basis the sub-function space has 155 distinct function de-

Sub-function \ mod.	Student											Freq.
	1	2	3	4	5	6	7	8	9	10	11	
import EE	1	1	1	1	1	1	1	1	1	1	1	1
transfer EE	1		1	1		1		1	1	1	1	0.73
distribute EE	1							1				0.18
guide EE		1										0.09
store EE								1				0.09
actuate EE					1	1			1	1		0.36
regulate EE		1	1		1	1	1	1		1		0.64
convert EE to TE	1	1	1	1	1	1	1	1	1	1	1	1
import solid	1	1	1	1	1	1	1	1	1	1	1	1
stabilize solid		1	1									0.18
position solid							1					0.09
guide solid	1		1	1	1	1	1	1	1	1	1	0.82
secure solid	1				1		1	1	1			0.45
transfer solid		1							1			0.18
store solid				1		1					1	0.27
release solid					1				1			0.18
separate solid									1			0.09
export solid	1	1	1	1	1	1	1		1		1	0.82
distribute TE										1		0.09
regulate TE											1	0.09
guide TE				1	1							0.18
transfer TE	1	1	1	1	1	1	1	1	1	1	1	1
stop TE		1							1		1	0.27
transfer solid2	1											0.09
store solid2	1			1	1	1		1	1		1	0.64
guide solid2				1	1	1						0.27
export solid2	1	1	1	1	1	1		1	1	1	1	0.82
import HE	1	1	1		1	1	1	1	1	1	1	0.82
transmit HE			1		1	1					1	0.36
guide HE	1											0.09
convert HE to ME	1		1			1	1	1	1	1	1	0.73
change ME						1						0.09
store ME	1	1	1			1	1	1	1	1	1	0.82
supply ME						1					1	0.18
transfer ME	1						1	1	1			0.36
import HH	1	1	1			1	1	1	1	1	1	0.73
guide HH	1										1	0.18
export HH											1	0.09
regulate control	1											0.09
guide ME		1										0.09
indicate status			1						1	1		0.27
sense status			1							1	1	0.27
measure TE						1				1		0.18
display status							1					0.09
distribute ME									1			0.09
convert ME to AE										1		0.09
change control											1	0.09

Figure 11. The sub-function space for toaster functional basis.

matrix is sparse and repetition of the function to function connections does occur.

Product Architecture Analysis

For the last part of the iterative experiment analysis, we briefly look at product architecture definition with the derived toaster functional models. Applying the module heuristics of Section 3.3,

Sub-function \ mod.	Student					Freq.
	1	2	3	4	5	
import EE	1	1	1	1		0.8
store EE	1	1	1	1	1	1
supply EE			1	1	1	0.6
regulate EE		1	1		1	0.6
actuate EE	1		1		1	0.6
transfer EE	1	1	1		1	0.8
convert EE to ME	1	1	1	1	1	1
change ME			1	1	1	0.6
guide ME	1				1	0.4
transfer ME			1		1	0.4
rotate solid					1	0.2
import HH	1	1	1	1	1	1
secure weight	1	1				0.4
stabilize weight		1				0.2
position weight		1				0.2
import solid	1	1	1	1	1	1
store solid	1	1		1		0.6
secure solid		1	1		1	0.6
couple solid			1		1	0.4
separate solid					1	0.2
indicate status		1	1			0.4
import HE		1	1	1	1	0.8
transfer HE			1			0.2
convert HE to ME		1	1	1		0.6
distribute ME		1		1	1	0.6
stabilize ME		1				0.2
secure ME		1				0.2
stop ME					1	0.2

Figure 12. The sub-function space for the power screwdriver functional basis.

Sub-function	Student				Freq.
	1	2	3	4	
actuate air			1		0.25
convert HE to ME				1	0.25
convert HE to PE	1	1	1		0.75
convert ME to PE				1	0.25
convert PE to ME		1	1	1	0.75
couple solid	1				0.25
export solid	1	1		1	0.75
guide solid			1		0.25
import gas	1		1		0.5
import HE		1	1	1	0.75
import HH	1	1	1	1	1
import solid	1	1	1	1	1
import weight		1		1	0.5
secure solid	1	1			0.5
secure weight		1		1	0.5
stabilize weight		1			0.25
store gas	1		1		0.5
store ME				1	0.25
store PE	1				0.25
store solid	1		1	1	0.75
supply gas	1				0.25
supply ME				1	0.25
supply PE	1				0.25
transfer PE	1				0.25

Figure 13. The sub-function space for the dart gun functional basis.

Sub-functions	Student											Freq.
	1	2	3	4	5	6	7	8	9	10	11	
accept signal											1	0.083
activate energy			1									0.083
actuate EE						1			1			0.250
adjust amount of energy								1				0.083
adjust heat to setting											1	0.083
adjust signal												0.083
adjust toast level	1											0.083
allow setting adjustment											1	0.083
balance weight			1									0.083
capture crumbs	1											0.083
channel crumbs	1				1							0.167
channel energy					1							0.083
collect crumbs		1										0.167
contain bakery product						1						0.083
contain crumbs					1	1						0.167
control EE						1						0.083
controller				1								0.083
convert bread to toast												0.083
convert displacement to heat												0.083
convert EE to heat	1			1	1		1		1			0.500
convert EE to ME									1			0.083
convert EE to TE		1							1			0.167
convert energy			1									0.083
convert HE to KE												0.083
convert HE to ME									1			0.083
convert HE to PE												0.083
convert heat to CE									1			0.083
convert heat to displacement												0.083
convert PE to KE												0.083
convert signal to target displacement												0.083
convert to ME						1						0.083
convert to motion						1						0.083
convert to regulator			1									0.083
convert to TE						1					1	0.167
determine settings							1					0.083
direct crumbs				1								0.083
direct EE		1										0.083
direct heat				1								0.167
direct ME		1										0.083
direct toast				1								0.083
discharge crumbs					1							0.083
discharge energy			1									0.083
discharge/ guide bread			1									0.083
disconnect EE												0.083
display settings							1					0.083
display target			1									0.083
dispose of crumbs			1									0.083
...												

Figure 14. A fragment of the sub-function space for the toaster without using the functional basis language.

a total of eight distinct modules are identified across all functional models. Of these eight modules, five had a frequency greater than 70% and three out of the eight appeared in every functional model. The most commonly used heuristic for identifying modules is the dominant flow proposition, while the convert-transmit proposition identified two modules. The high frequency with which modules were identified offers further support of the consistency of the derived functional models, as the heuristics require similar models to produce similar modules. The lack of modules identified by the branching flow heuristic indicates to us that our concept of parallel function chains needs some refinement and will be addressed in future work.

5 FUTURE EXPERIMENTAL WORK

We have presented several functional modeling experiments focused on measuring the repeatability of functional modeling of students. Currently, we are extending and formalizing the functional modeling experiment to include both industry and academia participants. The experiment design follows the engineering design research method proposed by Antonsson (1987) and Dixon (1988). Our hypothesis is that the functional basis derivation method will produce repeatable functional models among different designers, given the same customer needs and process choices. In order to test our hypothesis, we propose the following experimental plan:

1. Prepare a functional modeling methodology that can be used to create function structures.
2. Have designers learn the functional basis modeling methodology.
3. Have designers apply the methodology on original and existing product designs.
4. Compare the functional models, and either validate or disprove our hypothesis.

Implementation of the experimental plan will involve three separate experiments. The first experiment consists of giving the test subject an existing product and a set of customer needs. The test subject is then asked to develop a functional model using any function modeling technique that they are familiar with.

The second experiment asks the subject to once again develop a functional model for the same product given in the first experiment. In this experiment, however, they are asked to use the functional model derivation methodology to

Figure 15. The frequency adjacency matrix for the toaster functional models.

develop their functional model for the product. The test subjects are given a user's manual on the functional model derivation methodology in order to learn the approach.

In the third and final experiment, the subject is given a set of customer needs and a new design problem and asked to develop a function structure for this new product. Product specifications will be given in order to constrain the process choices in the functional model for comparison purposes. In this experiment the test subject is asked to design a power supply for a radio in which human mechanical energy is stored and delivered as electrical energy. Once again the subjects are asked to use the functional model derivation methodology to create their functional model for this product.

In preliminary trials, the three experiments have been given to approximately 10 college students. These trials have given promising results and very useful suggestions. Even with a limited knowledge of functional modeling, most of the students are able to complete experiment one without any difficulty. In the second experiment, we have seen some difficulty with the time required in learning the functional basis methodology. This trial run on students has shown many ways on which to improve the experiment before we run it on industry and academia.

6 IMPACT: CONCLUSIONS AND FUTURE DIRECTIONS

The results, analysis, and discussion of our early functional-modeling experiments are very encouraging. The basic elements within a functional-modeling theory are enumerated and applied within a variety of classroom settings. Three products are used as case studies to gather raw data of functional modeling efforts from both undergraduate and graduate students. The backgrounds of these students range from beginning product designers to experts that have designed products for over twenty years.

Quantitative analyses are applied to the raw data, in addition to qualitative assessments of the cumulative students' work. Frequencies of function occurrence are calculated before and after functional models are converted with a common functional basis. The topologies of the functional models are also assessed with the new concept of adjacency matrices for functional modeling.

The analyses provide insights into the current state of the theoretical elements of functional modeling. In general, the results show very good repeatability of function usage, even in the context where many of the students are cognitively digesting and applying functional analysis techniques for the first time. They also demonstrate that the topological layout and connections of the functions are consistent. This consistency is essential if the underlying physics of the product flows are to be reasonably represented in the models.

Differences in the student results arise from two primary sources. The first entails a different range of customer needs (where the basic subsets are identical) and different process choices being applied by the student. These differences are a positive consequence of functional modeling. They provide multiple and equally valid approaches to establish the function of a product.

These approaches, in turn, will lead to a greater number of feasible architectures, portfolios, and form solutions of a product. The second source is misapplication or errors in the execution of the functional modeling techniques. In isolated cases, students misapplied the common basis functions, connected functions inconsistently, or did not understand the underlying physics of the processes they were modeling. This second source identifies weaknesses in the current functional modeling theories, where critical advancements are needed (such as the improved ability to represent control schemes and the signal flows within an overall product function).

Even though the statistical sample size of the experiments is not large, these insights clearly show that functional modeling has good repeatability. Functional models of products can be systematically generated with sound outcomes. The insights also show that weaknesses exist in our functional modeling approach. Enhancements and further fundamentals are needed to address the weaknesses.

The next step in this research is to perform a more comprehensive study of functional modeling. The range and type of subject participating in the experiment must be expanded. We discuss, in the body of the paper, a more extensive experimental procedure to address this need. The results from this study should identify, more completely, the current repeatability of functional modeling and the existence of prevalent weaknesses.

Once the more extensive experiments are completed, a similar analysis will be performed on the raw data. Alternative analyses will also be considered. These analyses should clearly indicate the needed evolution in the theoretical underpinnings of functional analysis. We will execute this evolutionary step to expand the theory, while balancing it with the efficacy of using the theory in actual product design applications.

REFERENCES

- Antonsson, E., 1987, "Development and Testing of Hypotheses in Engineering Design Research", *Journal of Mechanisms, Transmissions, and Automation in Design*, 109/153.
- Akiyama, K. Function Analysis: Systematic Improvement of Quality Performance, Productivity Press, 1991.
- Altshuller, G., 1984, *Creativity as an Exact Science*, Gordon and Breach Publishers.
- Collins, J., Hagan, B., and Bratt, H., 1976, "The Failure-Experience Matrix - a Useful Design Tool," *Transactions of the ASME, Series B, Journal of Engineering in Industry*, 98:1074-1079.
- Cutherell, D., 1996, "Chapter 16: Product Architecture," *The PDMA Handbook of New Product Development*, M. Rosenau Jr. et al., ed., John Wiley and Sons.
- Dixon, J., 1988, "On Research Methodology Towards A Scientific Theory of Engineering Design", *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing*, 1/3.
- Hubka, V. and Ernst Eder, W., 1984, *Theory of Technical Systems*, Springer-Verlag, Berlin.
- Hundal, M., 1990, "A Systematic Method for Developing Func-

- tion Structures, Solutions and Concept Variants,” *Mechanism and Machine Theory*, 25(3):243-256.
- Iwasaki, Y., M. Vescovi, R. Fikes, and B. Chandrasekaran, 1995, “Casual Functional Representation Language with Behavior-Based Semantics,” *Applied Artificial Intelligence*, 9:5-31.
- Kirschman, C., Fadel, G., 1998, “Classifying Functions for Mechanical Design,” *Journal of Mechanical Design, Transactions of the ASME*, 120(3):475-482.
- Koch, P., Peplinski, J., Allen, J. and Mistree, F., 1994, “A Method for Design Using Available Assets: Identifying a Feasible System Configuration,” *Behavioral Science*, 30:229-250.
- Lai, K., and W. R. D. Wilson, 1989, “FDL - A Language for Function Description and Rationalization in Mechanical Design,” *Journal of Mechanics, Transmissions, and Automation in Design*, 111:117-123.
- Malmqvist, J., Axelsson, R., and Johansson, M., 1996, “A Comparative Analysis of the Theory of Inventive Problem Solving and the Systematic Approach of Pahl and Beitz,” *Proceedings of the 1996 ASME Design Engineering Technical Conferences*, 96-DETC/DTM-1529, Irvine, CA.
- McAdams, D., Stone, R., and Wood, K., 1998, “Product Similarity Based on Customer Needs,” *Proceedings of DETC98*, DETC98/DTM-5660, Atlanta, GA.
- Miles, L., 1972, *Techniques of Value Analysis Engineering*, McGraw-Hill.
- Murdock, J., Szykman, S. and Sriram, R., 1997, “An Information Modeling Framework to Support Design Databases and Repositories,” *Proceedings of DETC’97*, DETC97/DFM-4373, Sacramento, CA.
- Otto, K. “Forming Product Design Specifications,” in *Proceedings of the 1996 ASME Design Theory and Methodology Conference*, Irvine, CA, 1996.
- Otto, K. and Wood, K., 1997, “Conceptual and Configuration Design of Products and Assemblies,” *ASM Handbook, Materials Selection and Design*, Vol. 20, ASM International.
- Otto, K. N. and K.L. Wood (2000), *Product Design: Techniques in Reverse Engineering, Systematic Design, and New Product Development*, Prentice-Hall, NY.
- Pahl, G. and Beitz, W., 1996, *Engineering Design: A Systematic Approach*, Springer-Verlag.
- Pimmler, T. and Eppinger, S., 1994, “Integration Analysis of Product Decompositions,” *Proceedings of the ASME Design Theory and Methodology Conference*, DE-Vol. 68.
- Schmidt, L. and Cagan, J., 1995, “Recursive Annealing: A Computational Model for Machine Design,” *Research in Engineering Design*, 7:102-125.
- Shimomura, Y., Tanigawa, S., Takeda, H., Umeda, Y., Tomiyama, T., 1996, “Functional Evaluation Based on Function Content,” *Proceedings of the 1996 ASME Design Theory and Methodology Conference*, 96-DETC/DTM-1532, Irvine, CA.
- Stone, R., Wood, K., Crawford, R., 1998, “A Heuristic Method to Identify Modules from a Functional Description of a Product,” *Proceedings of DETC98*, DETC98/DTM-5642, Atlanta, GA.
- Stone, R., K. Wood and R. Crawford, 1999a, “Using Quantitative Functional Models to Develop Product Architectures,” To Appear *Design Studies*.
- Stone, R., K. Wood and R. Crawford, 1999b, “Product Architecture Development with Quantitative Functional Models,” *Proceedings of DETC99*, DETC99/DTM-8764, Las Vegas, NV.
- Stone, R., Wood, K., Crawford, R., 2000, “A Heuristic Method for Identifying Modules for Product Architectures,” *Design Studies*, 21(1):5-31.
- Stone, R., Wood, K., 1999, “Development of A Functional Basis For Design”, *Proceedings of the ASME Design Theory and Methodology Conference*, DETC99/DTM-8765, Las Vegas, NV
- Szykman, S., Racz, J., Sriram, R., 1999, “The Representation of Function in Computer-Based Design”, *Proceedings of the ASME Design Theory and Methodology Conference*, DETC99/DTM-8742, Las Vegas, NV
- Ullman, D., 1997, *The Mechanical Design Process 2nd ed.*, McGraw-Hill.
- Ulrich, K. and Eppinger, S., 1995, *Product Design and Development*, McGraw-Hill.
- Umeda, Y. and T. Tomiyama, 1997, “Functional Reasoning in Design,” *IEEE Expert Intelligent Systems and Their Applications*, 12(2):42-48.
- Urban, G. and Hauser, J. 1993, *Design and Marketing of New Products*, Prentice Hall, New York, 2nd edition.
- VAI (Value Analysis Incorporated), *Value Analysis, Value Engineering, and Value Management*, Clifton Park, New York, 1993.
- Zamirowski, E., Otto, K., 1999, “Identifying Product Portfolio Architecture Modularity Using Function and Variety Heuristics,” *Proceedings of the ASME Design Theory and Methodology Conference*, DETC99/DTM-8760, Las Vegas, NV.